Spectral Envelope Modelling for Full-Band Speech Coding

Chamran Moradi Ashour

School of Electrical Engineering

Thesis submitted for examination for the degree of Master of Science in Technology.

Espoo 21.11.2016

Thesis supervisor:

Prof. Paavo Alku

Thesis advisors:

Prof. Tom Bäckström

Dr. Guillaume Fuchs

Aalto University
School of Electrical
Engineering

Acknowledgment

I would like to thank Prof. Paavo Alku for his cooperation. I would like to express my special appreciation and thanks to my Prof. Tom Bäckström for giving me the opportunity to make the master thesis with him and for his positive energy, encouragement, valuable ideas and consistent support.

I would also like to express my deepest gratitude to my advisor Dr. Guillaume Fuchs for his continuous support and guidance throughout the thesis. His knowledge, patience and valuable advice did much to bring this work to a successful conclusion.

I am thankful to my parents for their dedication and many years of support.

Chamran Ashour

AALTO UNIVERSITY
SCHOOL OF ELECTRICAL ENGINEERING

ABSTRACT OF THE
MASTER'S THESIS

Author: Chamran Moradi Ashour

Title: Spectral Envelope Modelling for Full-Band Speech Coding

Date: 21.11.2016 Language: English Number of pages: 9+61

Department of Signal Processing and Acoustics

Professorship: Speech Communication Technology

Supervisor: Prof. Paavo Alku

Advisors: Prof. Tom Bäckström, Dr. Guillaume Fuchs

Speech coding considering historically narrow-band was in the latest years significantly improved by widening the coded audio bandwidth. However, existing speech coders still employ a limited band source-filter model extended by parametric coding of the higher band. In this thesis, a full-band source-filter model is considered and especially its spectral magnitude envelope modelling.

To match full-band operating mode, we modified, tuned and compared two methods, Linear Predictive Coding (LPC) and Distribution Quantization (DQ). LPC uses autoregressive modeling, while DQ quantifies the energy ratios between parts of the spectrum. Parameters of both methods were quantized with multi-stage vector quantization. Objective and subjective evaluations indicate the two methods used in a full-band source-filter coding scheme perform on the same range and are competitive against conventional speech coders requiring an extra bandwidth extension.

Keywords: Spectral Envelope Modelling, Linear Predictive Coding, Distribution Quantization, Moving Average Multi-Stage Vector Quantization, Full-Band Speech Coding

List of Figures

List of Tables

Abbreviations

SNR	Signal to Noise Ratio
MSE	Mean Squared Error
SD	Spectral Distortion
PESQ	Perceptual Evaluation of Speech Quality
POLQA	Perceptual Objective Listening Quality Assessment
MUSHRA	MUltiple Stimuli with Hidden Reference and Anchor
LP	Linear Prediction
LPC	Linear Predictive Coding
LSF	Line Spectral Frequency
DQ	Distribution Quantizer
AbS	Analysis-by-Synthesis
CELP	Code-Excited Linear Prediction
ACELP	Algebraic Code-Excited Linear Prediction
RCELP	Relaxed Code-Excited Linear Prediction
CS-ACELP	Conjugate-Structure Algebraic Code-Excited Linear Prediction
IIR	Infinite Impulse Response filter
FIR	Finite Impulse Response filter
VQ	Vector Quantization
MS-VQ	Multi Stage Vector Quantization
MA-MSVQ	Moving Average-Multi Stage Vector Quantization
DFT	Discrete Fourier Transform
AMR-WB	Adaptive Multi-Rate Wideband
EVS	Enhanced Voice Services
SBR	Spectral Band Replication
USAC	Unified Speech and Audio Coder
xHE-AAC	Extended High Efficiency Advanced Audio Coding
MPEG	Moving Picture Experts Group
WB	Wide-Band
FB	Full-Band
BW	Bandwidth

Contents

1 Introduction

Nowadays, people in any corner of the world are able to have speech communication independent from space and time. This achievement in communications would not be feasible without appropriate speech coding algorithms. Speech coding first become possible after *Digital Revolution* age and has continued its evolutionary progress to the point where we are now. There is no concern of accessibility of speech communications anymore, but instead *quality* and *cost* are into consideration. In speech coding, criteria for evaluation of *cost* are efficiency, complexity, bit rate and memory requirement. Any of these criteria can then appear in different forms such as energy consumption, bandwidth consumption and end-user cost. Regarding billion users of mobile communications, smallest improvement in speech coding algorithm could have significant effect on the cost. Quality, on the other hand, is defined as *perceptual transparency* [7]. Although several objective measures such as Log Spectral Distortion, PESQ [16] and POLQA [19] have been introduced to evaluate the quality of a speech codec, none of them are as well-founded as subjective listening test. The ideal quality is achieved when the perceived speech signal can not be distinguished from the original one. Considering above-mentioned facts, we can point that the objective of speech coding technology has turned from enabling speech communication independent from the place, into achieving the highest quality of speech communication with the least cost. Any improvement on speech codecs necessitates solid knowledge of speech production procedure.

1.1 Speech Production

A speech signal is produced in the following steps. First, lungs thrust the air out. The air flows towards larynx where vocal folds are located. Finally, in the vocal tract (pharynx, velum, oral cavity, nasal cavity, lips, tongue, etc.) the airflow is shaped into perceptually speech signal [4].

Speech production procedure can be expressed as source-filter model. Sounds are produced when the source (lungs and larynx) *excites* the filter (vocal tract). When the vocal tract is excited by the source, different shape of vocal tract results in different resonances which are known as formant frequencies. Formant frequencies are very important properties of speech signal and contain the information that human's ear requires to distinguish between different phonemes of speech. In case of consonants, the airflow is restricted in the vocal tract by lips, teeth or tongue, while for vowels no restriction is applied on the airflow [23].

Sounds can be also categorized into voiced and unvoiced sounds. In voiced signals, vocal folds are oscillating with a certain frequency which is known as pitch. This oscillation of vocal folds causes to have semi-periodic excitation. In unvoiced speech the source is aperiodic and cause noisy turbulences at contraction of the vocal tract. Figure 1.1 depicts the general mechanism of the speech production. Velum is

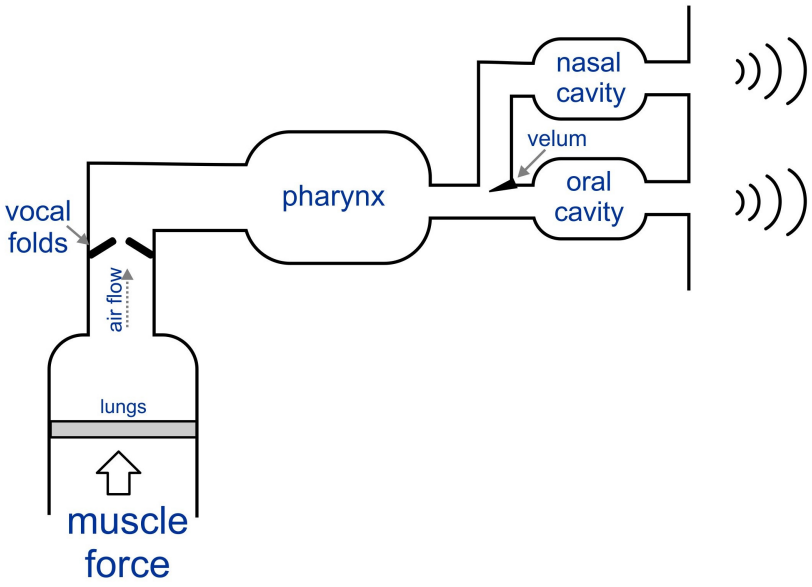


Figure 1.1: Speech production mechanism.

responsible for switching between oral cavity and nasal cavity. However, in vocal tract modelling, nasal cavity is omitted [7].

1.2 Envelope Modelling in Speech Coding: Prior Art

Speech signal is perceptually highly redundant. Moreover, human's ear has psychoacoustic characteristics which means it is more sensitive to low frequency components than to high frequency ones. Spectral envelope modelling can exploit the redundancies and at the same time consider psychoacoustic characteristics of the human's ear. This property of spectral envelope has enabled speech codecs to code the speech signal with high quality and low bitrate. Spectral envelope smoothly links the peaks in magnitude spectrum plane of a signal. The spectral envelope shape show a succession of valleys and hills which represent the formants of the speech.

Among the techniques of spectral envelope modelling, linear predictive coding (LPC) is the most common one which has been widely used in speech codecs. In the source-filter model, filter represents the vocal tract. Vocal tract can be considered an acoustic filter which takes the source signal and converts it into perceptually speech signal. This acoustic filter (vocal tract) can be approximated very efficiently by concatenation of successive, straight, round, lossless and piece-wise constant radius tubes [7]. The concatenation of the successive tubes leads to a lattice-form filter structure which can be expressed as linear predictive filter [7]. Figure 1.2 shows the tube model of the vocal tract.

LPC is highly efficient and outstandingly accurate for voice (matches source-filter

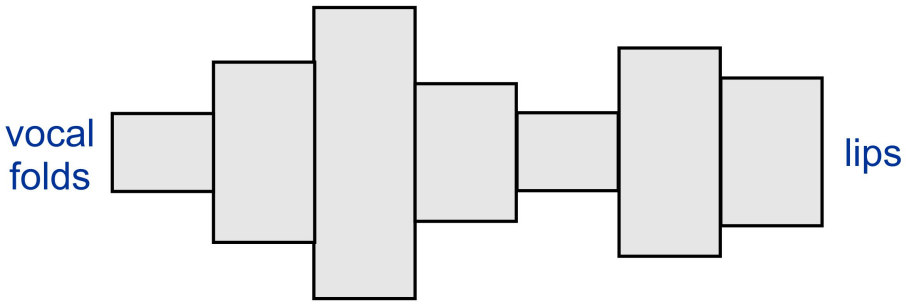


Figure 1.2: Tube model of vocal tract.

model). It has also a reliable physical interpretation (resonance representation). LPC envelope modelling was used in Code-excited linear prediction (CELP) speech algorithm [31]. CELP was originally designed for narrow-band (sampling rate 8 kHz) and adapted to wide-band (sampling rate 12.8 kHz) [2], [1]. However, LPC technique can show limitations when being extended to a large audio bandwidth. Increasing the order will automatically increase its complexity and makes even more difficult the realization of the vector quantization of its parameters. Moreover, LPC suffers from frequency resolution inaccuracy especially at lower frequencies which becomes more prominent for full-band speech. Finally, the numerical stability of the Levinson-Durbin and Chebychev algorithms required in LPC technique can become problematic for high orders of the system.

Distribution Quantization (DQ) is one of the recent envelope modelling technique which can be an alternative to LPC in the speech codecs. DQ is based on distribution of the spectral mass of a signal [20]. The strong point of DQ is that its parameters are orthogonal and uncorrelated with each other. This property results in significantly low computational complexity. In wide-band speech coding, DQ technique performs as good as LPC with considerably lower complexity [24]. However DQ was never used for modelling a spectral envelope of a super-wideband or full band speech signal, which we propose in the present work. The envelope modelling of speech and its usage within speech coding is further explained and discussed in chapter 2.

1.3 Contributions

In this thesis, we address the problem of modelling and coding the spectral envelope of super wide-band and full-band speech by considering both LPC and DQ techniques. The final objective is to improve the quality of speech coding by extending the source-filter model to cover an audio bandwidth up to 20 kHz. In case of LPC, along with LPC parameters, we need to modify the weighting used in wide-band. We explain the logic behind LPC, an example of LPC in wide-band and how we extend it to full-band case in chapter 3. In DQ technique, unlike [24] we used fixed positioning scheme based on mel scale for DQ parameters. We discuss the background of DQ along with its extension to full-band employing mel scale in chapter 4. In the end

of chapter 3 and chapter 4 we evaluate each technique independently. Moreover, we compare performance of LPC to DQ using different objective and subjective measures. In chapter 2, we explain what measurement methodologies which we use for all assessments in the thesis. The comparison takes place in chapter 5. For having even condition for both approaches, we use same speech codec which is explained in chapter 2. We use vector quantization for both DQ and LPC. In chapter 2, we present the type of vector quantizer we use and we show how we extend the quantizer for full-band in chapter 2. we examine the orthogonality of DQ parameters in full-band by measuring SD of DQ with uniform quantization in the end of chapter 4.

Spectral modelling of full-band speech by LPC is introduced in Chapter 3. Starting from the configuration usually adopted for wide-band, we extend the model to a wider audio bandwidth. The order of the filter, the pre-emphasis filtering, the perceptual weighing as well as the quantization scheme design are justified through different measurements and optimization steps.

Chapter 4 is dedicated to the DQ technique and its extension to full-band coding. The main contribution is to propose for the first time to use a Vector Quantization (VQ) for coding its parameters. For high number of DQ parameters, we show that VQ has a slight advantage over entropy constrained Scalar Quantization (SQ). This contradicts the original hypothesis from previous works that DQ parameters are always uncorrelated, which seems only true for low orders. Adopting a Multi-Stage VQ with M-best search with the same bit allocation for both LPC and DQ is also a fair way to compare the two techniques.

The two techniques are compared to each other and within a complete full-band speech doing scheme in Chapter 5. Objective and subjective assessments were conducted. Listening test results show that the two techniques perform almost the same with a slight advantage of LPC over DQ. Compared to the state-of-the-art codec ISO/MPEG Unified Speech and Audio Coder (USAC) using a wide-band speech coder based on ACELP and Bandwidth extension based on SBR, the proposed full-band spectral modelling associated to a full-band source-filter model performs only slightly worse for clean speech while being on par for noisy speech. These results are encouraging, knowing that the system show a significant lower algorithmic delay and complexity compared to the conventional approach and is also only at the first stage of development.

2 Speech Coding and Vector Quantization

In this thesis, our aim is to compare the performance of two models for the spectral envelope of speech signals, namely Linear Predictive Coding (LPC) and Distribution Quantization (DQ). The two methods are compared to each other in a full-band configuration within the same framework based the well-established CELP coding paradigm, originally designed for narrow-band, adapted later to wide-band, and extended for this work to full-band speech coding. Moreover, the same vector quantization technique is employed in both cases and common objective measurements are used to assess their performance. In this chapter along with envelope modelling we briefly review state-of-the-art speech coding and vector quantization techniques used further in this work.

This chapter starts with an introduction to the concept of envelope modelling. This is followed by true envelope modelling which is used as a reference in the spectral distortion measurement. We proceed to describe the concept of vector quantization and more specifically the technique called Moving-Average-Multi Stage VQ (MA-MSVQ). Following this, we continue with the Analysis by Synthesis principle for speech coding at low bit-rates and its most successful variant ACELP. Finally we define the objective and subjective measurements used in the thesis.

2.1 Envelope Modelling

Spectral envelope is a curve which smoothly links the peaks in magnitude spectrum plane of a signal [33]. Due to its efficiency in representation of properties of a signal, it is widely used in audio and speech processing algorithms such as speech coding, audio enhancement, speech recognition and speech synthesis. In a speech signal, formant frequencies which are resonance frequencies of the vocal tract, contain the information that human's ear needs to distinguish between different phonemes of speech. The peaks in spectral envelope of a speech signal represent these formant frequencies. Figure 2.1, which is an example of a speech signal along with its estimated spectral envelope by LPC method, shows how spectral envelope gives the overall shape of the magnitude in frequency domain with required formants.

Spectral envelope of a speech signal can be estimated through several methods such as Linear Predictive Coding (LPC) [31], Distribution Quantization (DQ) [24], Discrete Cepstrum Spectral Envelope [13] and True Envelope [38].

In our case LPC and DQ which are fundamental tools of our work and True Envelope are used. We use True Envelope in the objective measurement Log Spectral Distortion as the ground truth reference to which we compare the obtained envelope models. We explain LPC and DQ in the next chapters in detail and True Envelope will be discussed briefly in the next section.

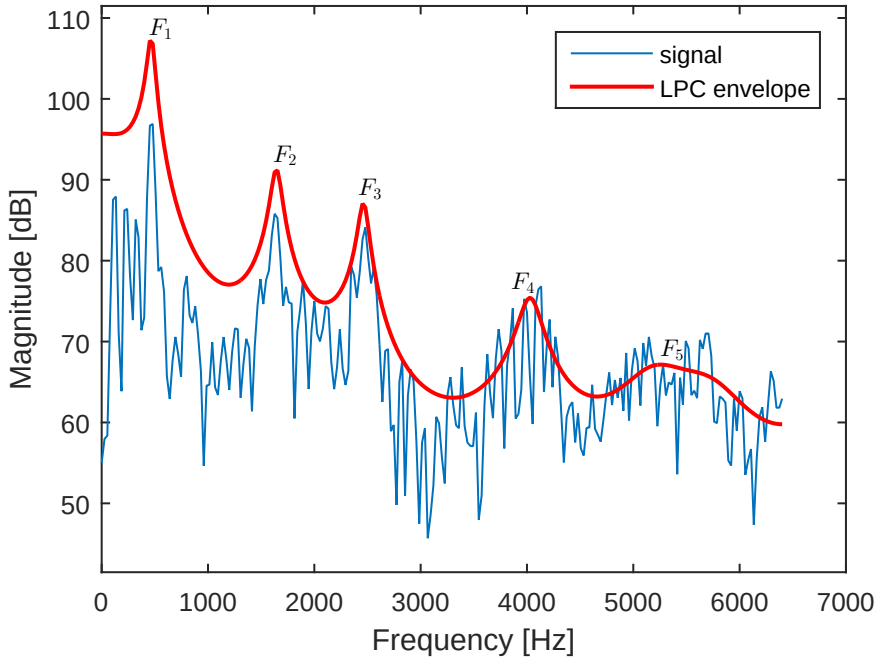


Figure 2.1: A frame of a speech signal with its LPC spectral envelope. F_1, F_2, F_3, F_4 represent formant frequencies.

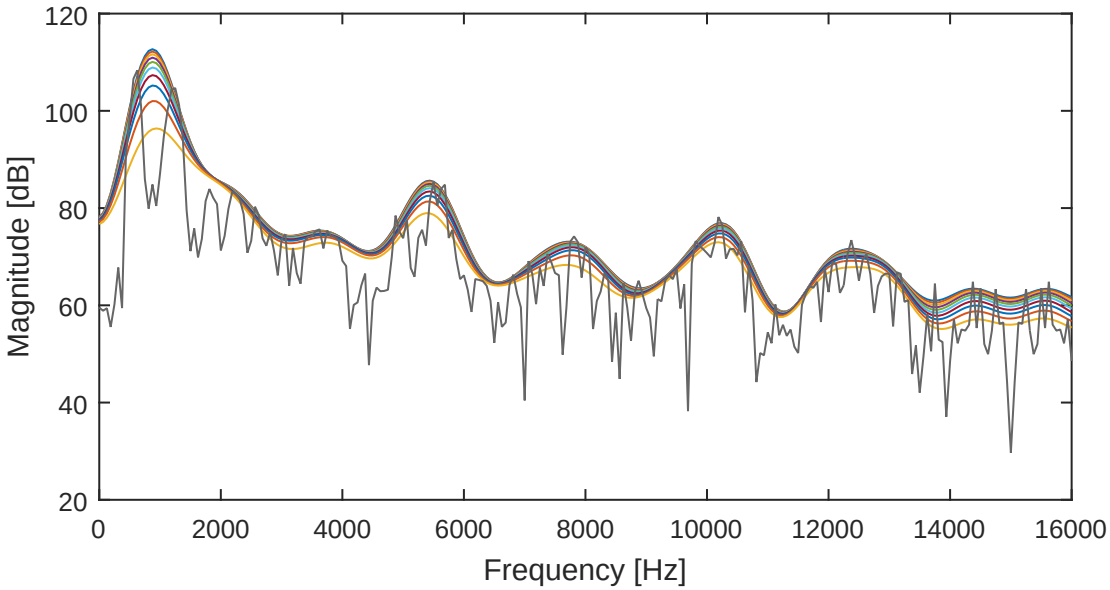


Figure 2.2: Speech signal with true envelope estimator with different number of iterations (1 to 10 iterations).

2.2 True Envelope Modelling

True envelope modelling is a cepstral based technique which gives a band limited envelope from a spectrum of a signal [9]. It uses an iterative procedure through which the smoothed spectral envelope of the input is updated with the maximum of itself and its cepstral representation. The iterative procedure can be mathematically defined as [37]:

$$A_i(k) = max(A_{i-1}(k), C_{i-1}(k)), \tag{2.1}$$

where k representing related frequency bin, $A_i(k)$ is the smoothed spectral of the current iteration i, A_{i-1} and C_{i-1} are respectively the smoothed spectral envelope and cepstral representation of A_{i-1} at iteration $i-1$. As an initialization value we have $A_0(k) = \log(|X(k)|)$ where $X(k)$ is the $k - point\ DFT$ of the framed signal $x(n)$.

Through this iterative procedure the valleys between the peaks of the spectrum is filled by the cepstral filter and the procedure is continued untill the whole spectrum is covered. The order of the cepstral filter -cepstral order- is defined as [38]:

$$\hat{O} = \frac{F_s}{2\Delta_F} = \alpha\frac{F_s}{F_0}, \alpha = 0.5, \tag{2.2}$$

where $\hat{O}$ is the cepstral order, F_s is sampling rate and F_0 is the fundamental frequency. Figure 2.2 shows the speech signal to which true envelope estimator with 10 iterations is applied. It was observed by visual inspections that the true envelope is a robust estimate of the spectral envelope and was not subject to systematic errors and order mismatch observed by LPC. Since the true envelope has per definition a variable order and is also defined in cepstral domain, the coding of its parameters is expected to be problematic and is therefore used only as a reference when comparing the different spectral envelope coding schemes.

2.3 Vector Quantization

One definition of quantization could be that quantization is a process of converting an infinite set of quantities to a finite set of quantities. In any quantization, there is a trade off between quantization quality in one side and complexity and memory requirement in the other side. In speech coding, there are two general techniques of quantization: scalar quantization where each sample of a set of discrete-time values is quantized separately and vector quantization (VQ) where a set of discrete-time values is considered in a vector and quantized jointly. VQ is a superior coding tool over scalar quantization . One of the main advantage is its ability to expoit the linear and non-linear depedence among the vector coordinates [15]. Another interesting advantage is the flexibility of its codebook size which can allocate a non-integer

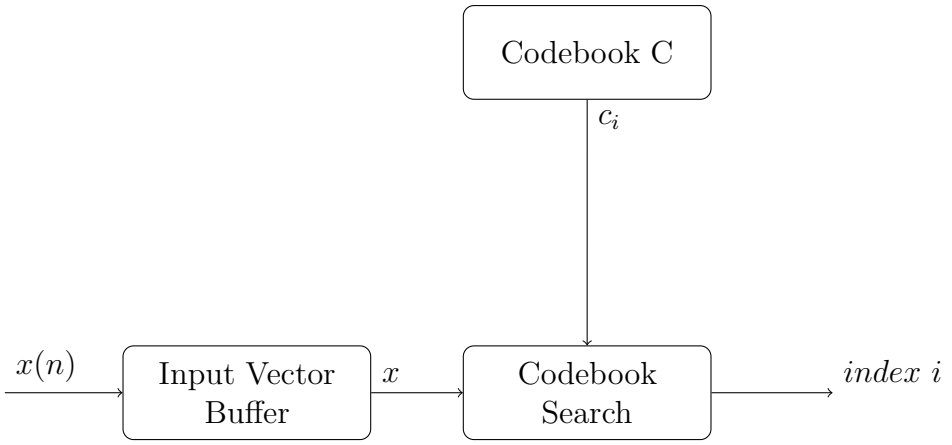


Figure 2.3: Block diagram of a general vector quantizer.

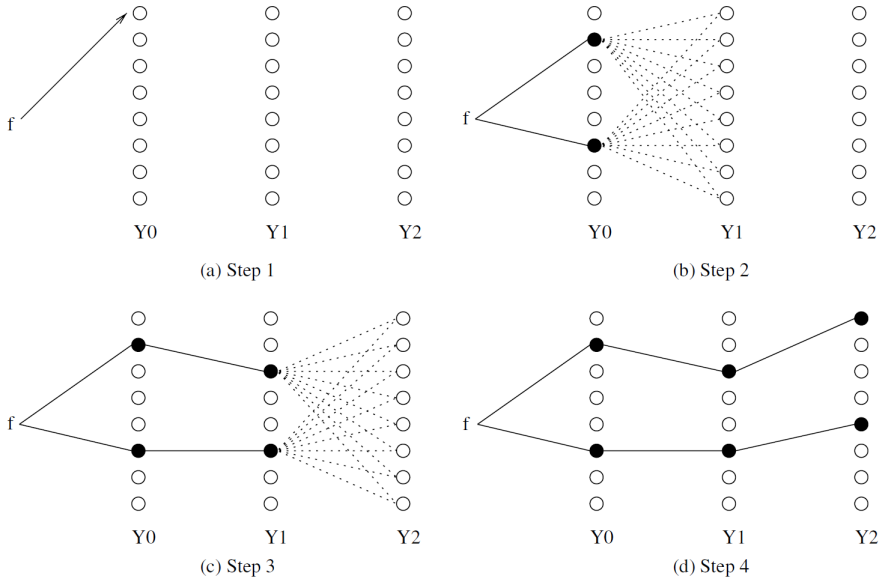


Figure 2.4: Steps in an M-best search [23].

number of bits per sample while scalar quantization is restricted to quantize each sample with an integer number of bits.

In VQ technique, a codebook is designed from which an entry is selected to represent an input vector a to quantize. An objective function is defined to be able to search the best match within the codebook. The objective function can be a Mean Squared Error (MSE) between each codevectors $c_i, 0 \leqslant i \leqslant L$ of the codebook of size L and the input vector or a distortion measure. In the spectral envelope coding, the spectral distortion (SD) is used as the objective function. The codevector for which the minimum SD is reached is selected to represent the input vector a. Only its index i needs to be transmitted since the codebook is shared between the encoder and decoder. Figure 2.3 shows a simple vector quantizer.

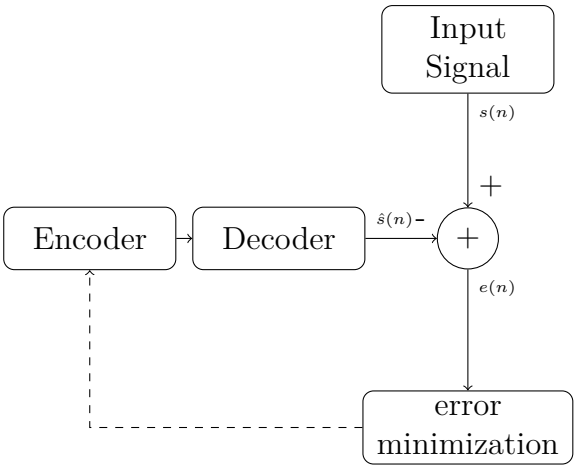


Figure 2.5: Block diagram of general analysis-by-synthesis [7].

With above-mentioned logic behind vector quantization, we can observe that the design of codebook plays the major role in the quantization scheme and is also the most complex part of the process. There are several techniques to design a codebook. Choosing the size and the constellation of the codebook is the first step and very essential. Different configurations of the codebook are possible and different technique can be found in the literature such as Split Vector Quantization, Merge Vector Quantization and Multi-Stage Quantization [7]. Then, the codebook needs to be trained off-line on a representative and large enough database. K-Means clustering is the usual algorithm for training the codebook [15]. Among these techniques, we retain MS-VQ for this work since it delivers the best performance if it is combined with a M-best search strategy [23]. The technique will be reviewed in detail in the following section.

2.4 MA-MSVQ

Moving Average Multi-Stage Vector Quantizer (MA-MSVQ) is one of the most common algorithm used as the quantization technique in speech coding. As mentioned before, complexity, memory requirement and efficiency are the criteria considered in designing a quantizer. MSVQ offers a quantizer with lower complexity and memory requirement than a single stage quantizer. As an example, a VQ coding, a vector with 25 bits will require $2^25\times$ size_of_vector words of memory in case of a single stage codebook, while in for example 4 stages quantizer with constellation [7 6 6 6] the memory requirement will decrease to: $2^7+2^6+2^6+2^6$ The algorithmic complexity will also also decrease with the same order of magnitude. This improvement is gained with cost of efficiency, but since for VQs with high bit rate the complexity of single stage VQ explodes that quantization becomes complexity-wise impractical.

Codebook Training in MSVQ

There are different methods to train the codebook in MSVQ. Among algorithms: sequential optimization, iterative sequential optimization and simultaneous joint codebook design, iterative sequential optimization is chosen due to the fact that it has the best trade off between computational cost and efficiency. In iterative sequential optimization, for each stage an initial codebook is chosen. Afterwards, for the codebook of each stage the quantization error is computed for all stages except the current one, and the training is used to get an updated version of the codebook of the current stage [23].

Searching Strategy

For quantizing a vector, codebooks are searched to find the nearest indices. To search codebooks there are several techniques among which an algorithm called M-best search is the most efficient one [23]. In M-best search algorithm M best paths from the first stage to the last stage of the MSVQ are explored. That is, in the first stage M best SD wise code-vectors and their quantization error are kept. In the next stage, codebook is searched M times to find M paths giving the lowest overall SD (also considering the SD obtained in the previous stage). This process is applied through all stages and in the end we will have M paths among which we choose the one with the lowest overall SD. Figure 2.4 is the example of this process with three stages each stage 8 bits and M equal to 2.

MA(Moving Average) in MA-MSVQ is a prediction of the current vector with the previous coded vector, which enhances the efficiency of the whole quantization scheme. For instance in LPC, the parameters of the spectral envelope can be represented by vectors of LSFs, whic are very highly correlated between sucessive frames over time. The quantizer of such a set of parameters can be improved by exploiting the similarities between successive vectors. This improvement can be achieved through prediction. That is, instead of quantizing an LSF vector, the difference between predicted vector of the LSF vector and the original LSF vector is quantized [23].

$$r_n = f_n - \hat{f}_n, \tag{2.3}$$

where f_n is the LSF vector and $\hat{f}_n$ is the prediction vector. The decoded LSF vector is then given by [23]:

$$\hat{f}_n = \hat{r}_n + \hat{f}_n, \tag{2.4}$$

where $\hat{r}_n$ is the quantized value of r_n. To do so, a prediction function in encoder and decoder side is needed. One possible prediction function is Moving Average (MA) function in which prediction is generated from the decoded codebook: $\hat{f}_n^k = \alpha_n \hat{r}_n^{k-1}$, where k_{th} set of LSF $\hat{f}_n$ is predicted. The decoded vector is then given by [23]:

$$\hat{f}_n^k = \hat{r}_n^k + \alpha_n \hat{r}_n^{k-1} \tag{2.5}$$

where α_n is the prediction constant which in our case is set to 0.33 for each n.

The optimal number of bits and the constellation of the codebook will be studied in the next chapter.

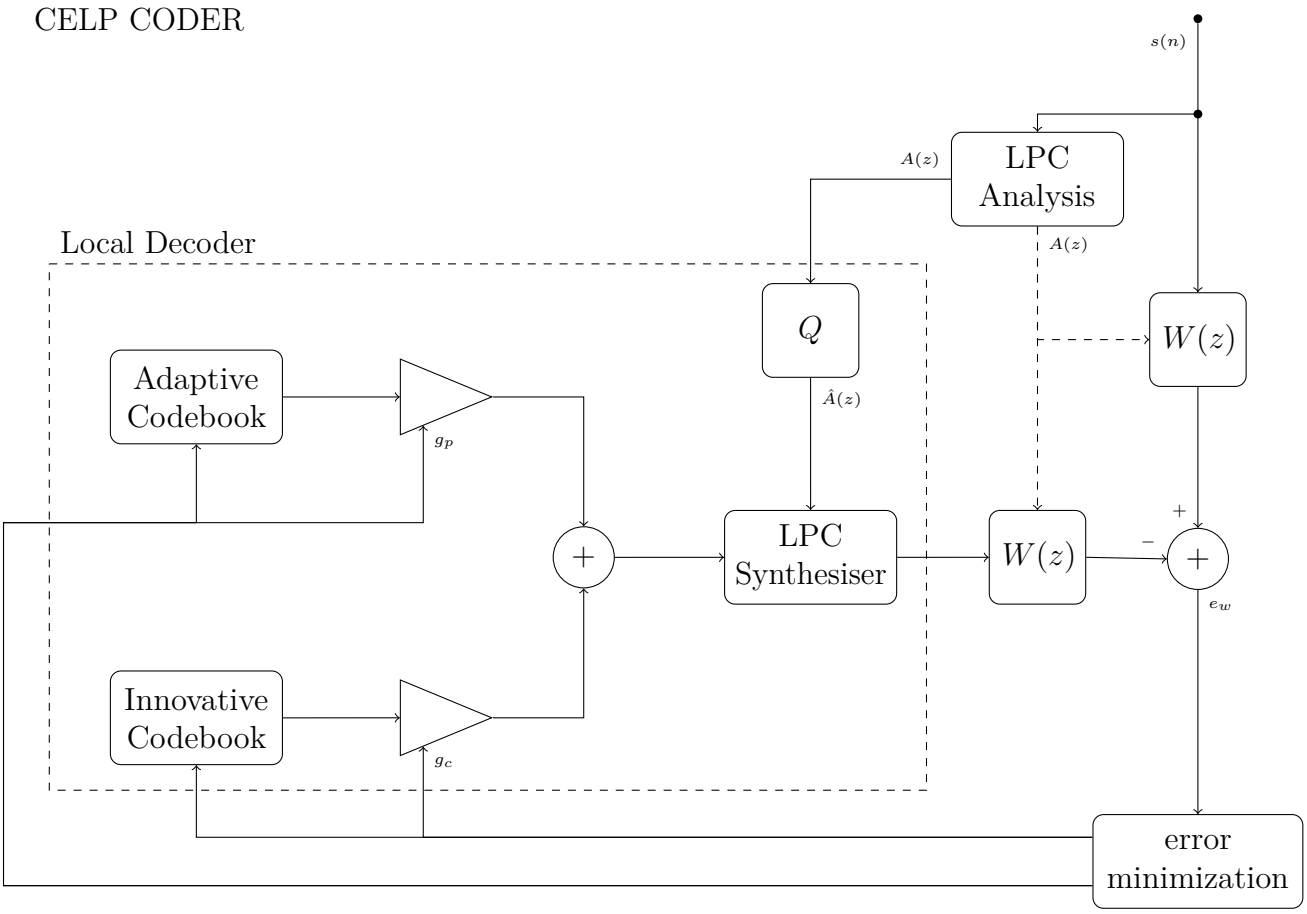
2.5 Analysis by Synthesis

In speech coding, Analysis-by-Synthesis (AbS) is a principle which is used to minimize waveform mismatch between the original speech signal and its synthesized version generated by a parametric coding of the signal relying on a filter-source model of the speech production. Figure 2.5 is a general AbS block diagram. In this figure block decoder simulates the synthesis side and the output of this block is compared with input signal. This process causes error and more specifically quantization error to be minimized. AbS is the fundamental idea of the speech algorithm CELP which has been developed into more efficient speech algorithms such as ACELP, CS-ACELP and RCELP. CELP algorithm has also been adopted to numerous communication standards such as G.719 [17] and AMR-WB [1].

2.6 ACELP

Algebraic Code-Excited Linear Prediction (ACELP) is a speech coding scheme based on CELP employing a specific algebraic structure for the innovative codebook, which allows a great complexity and storage saving. Figure 2.6 is a general block diagram of CELP/ACELP encoder and decoder. In block LPC Analysis, which is also known as shor-term prediction, linear spectral frequencies are computed. This block will be studied in the next chapter in detail. Q and $W(z)$ stands for quantizer and weighting filter respectively. In CELP/ACELP the algorithm is based on a closed-loop search within the adaptive codebook and innovative codebook by minimizing the coding error in a perceptually weighted domain. The weighting filter $W(z)$, which transform the signal into the perceptually weighted domain, has as effect to shape the coding noise so that it appears mostly in the frequency regions which are psychoacoustically less perceivable. The adaptive codebook is is used as a long-term prediction and exploits the main periodicity of the speech a.k.a pitch. Typically the fundamental frequency of speech lies between 375 and 55Hz, which corresponds to pitch lags 2.7 and 18ms respectively. By searching in adaptive codebook pitch lags and corresponding gain are found. The innovative codebook handles the components which are non predictable. As innovative codebook in CELP, stochastic codebook results in high complexity and memory requirement. That is the reason why, ACELP employs an algebraic codebook which allows a pratical and very efficient realisation of CELP by reducing drastically both algorithmic complexity and memory requirements.

ACELP codec can be represented in more detail as in Figure 2.7. The LPC function block is LP coefficients $A(z)$, quantize LP coefficients $\hat{A}(z)$ and coding the corresponding LSFs indices into the bitstream. In the next chapter, we will explain this block with related procedure in detail. Input speech signal $s(n)$ is passed through



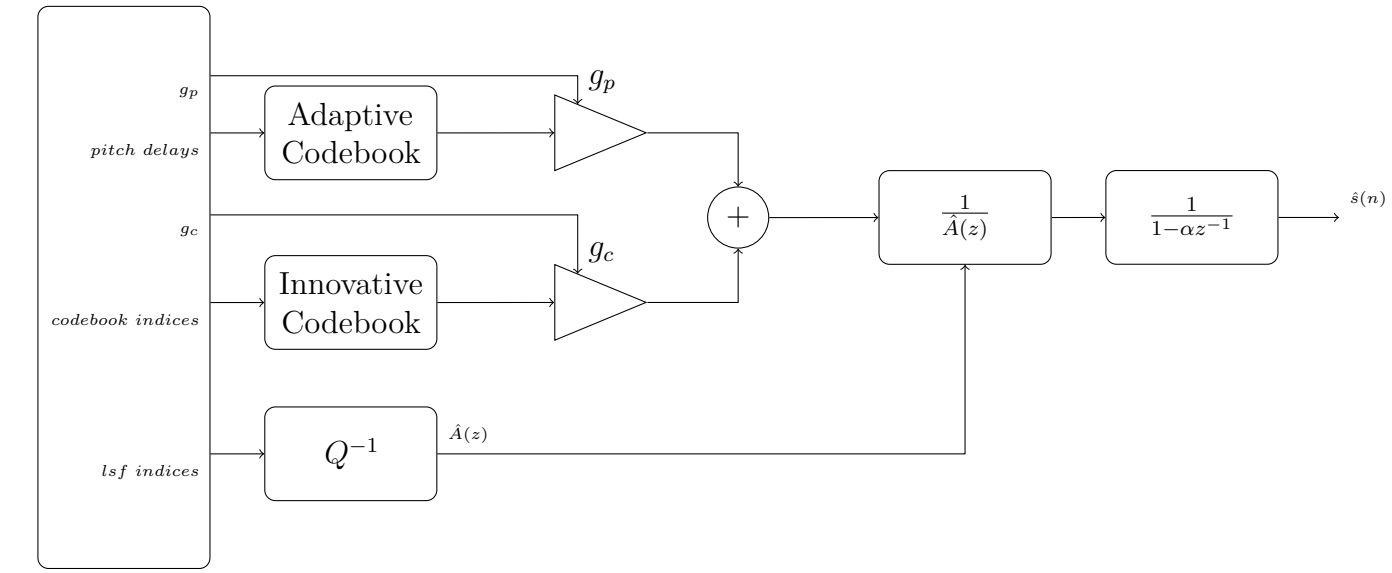


Figure 2.6: Block diagram of CELP/ACELP encoder along with decoder.

weighting filter $w(z) = \frac{A(z/\gamma 1)}{1-\alpha(z^{-1})}$ where $\gamma 1$ is constant between 0 and 1, and α is the pre-emphasis factor. The maximum of auto-correlation function of the weighted input $sw(n)$ is called open loop delay T_{op} and is used to limit the search range for the optimal pitch delay in the adaptive codebook analysis-by-synthesis search. This two stage optimization causes complexity of algorithm to decrease significantly. The IIR filter shown in block $Impulse \quad Response$ $(h(n))$ corresponds to the impulse response of the LPC synthesis filter followed by the perceptual weighting filter and is computed as $h(n) = w(z) * \frac{1}{\hat{A}(z)}$. In block $Target \quad Signal$ we compare the weighted input signal $sw(n)$ to zero input response $ZIR(n)$ to get the target signal $x(n)$ as $x(n) = sw(n) - ZIR(n)$. In block $Adaptive \quad Codebook \quad Search$ the optimal pitch delay and gain g_p are derived and conveyed for further encoding within the bitstream. The corresponding codevector $v(n)$ derived from the past decoded LP excitation and the pitch delay is used to update target signal $x(n) = x(n) - g_p * y(n)$.

In algebraic codebook, each vector is composed of a set of interleaved permutation codes which contains few nonzero elements. This only few nonzero elements and the fact that codebooks are generated algebraically, makes algebraic codebook significantly efficient in terms of memory requirement and complexity. Table 2.1 is an example of the structure of an algebraic codebook with 40-sample subframe and 5 tracks. The codebook vector $c(i)$ consists of five pulses in a possible 40-sample vector and all other locations are set to zero. These five pulses are computed as:

Table 2.1: 5-pulse algebraic codebook tracks for a 40-sample subframe [23].

Track	Pulse number	Possible locations
1	i_0	0,5,10,15,20,25,30,35
2	i_1	1,6,11,16,21,26,31,36
3	i_2	2,7,12,17,22,27,32,37
4	i_3	3,8,13,18,23,28,33,38
5	i_4	4,9,14,19,24,29,34,39

$$c(i) = s_0\delta(i - p_0) + s_1\delta(i - p_1) + s_2\delta(i - p_2) + s_3\delta(i - p_3) + s_4\delta(i - p_4) \tag{2.6}$$

where i is the number of sample in the subframe which in our example is $i = 0, .., 39$, s_i and p_i are the sign and position of the ith pulse respectively and δ represents unity pulse amplitude [23]. The best codebook vector, c_i at the index i is then decided by maximizing :

$$\Delta_i = \frac{\left(\sum_{n=1}^{N} d(n)c_i(n)\right)^2}{c_i^T H^T H c_i}, \tag{2.7}$$

where $d(n)$ is the correlation result of updated $x(n)$ and $h(n)$, H is the lower triangular Toepliz matrix with diagonals $h(0), h(1), h(2), ..., h(39)$ [6]. For more detail explanation of ACELP references [32] and [3] can be used.

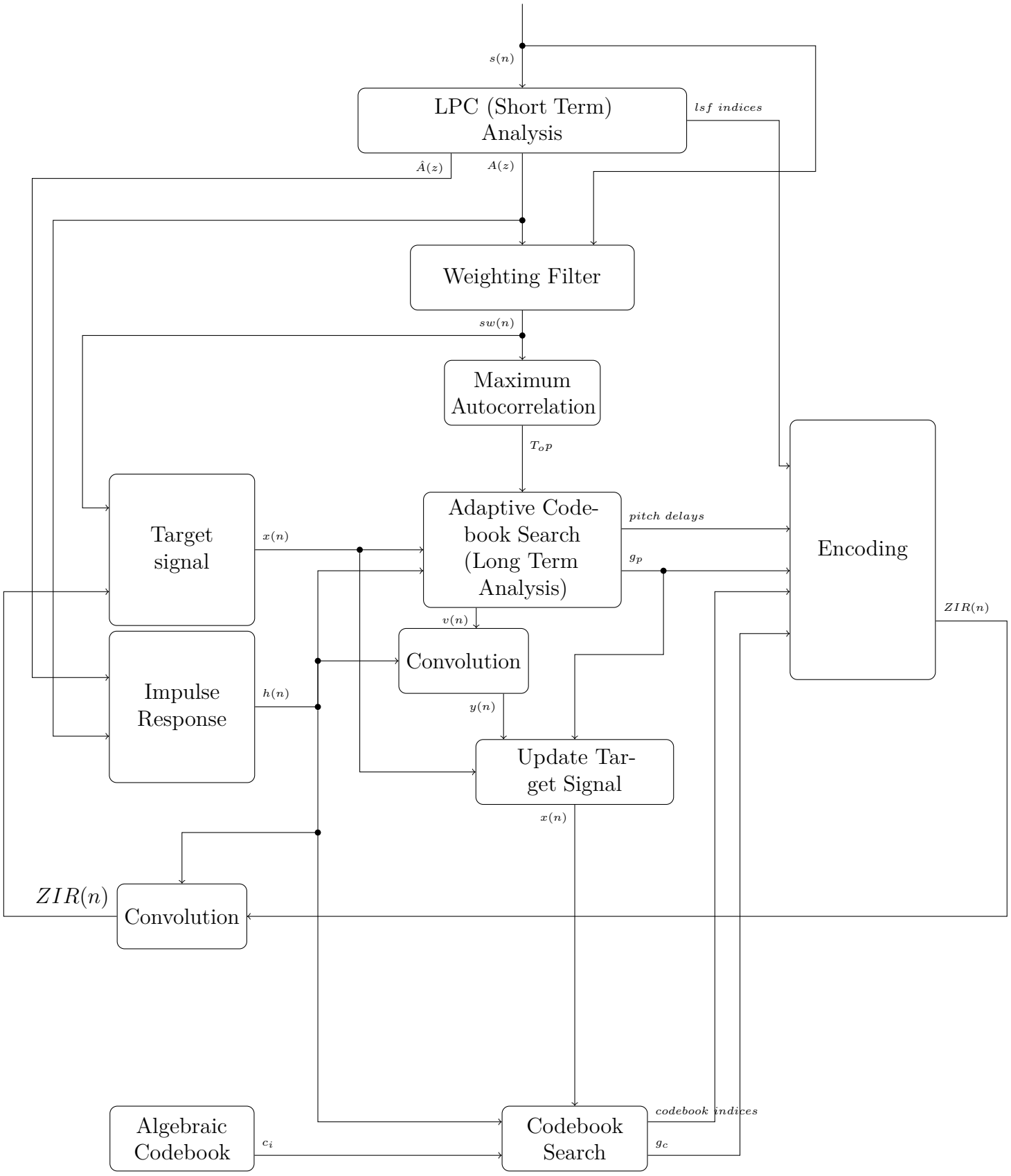


Figure 2.7: ACELP block diagram in detail.

2.7 Objetive and Subjective Measurements

As mentioned in the introduction, for evaluation of our implementations we used mainly SD as an objective measurement for direct evaluation of the spectral envelope modelling. For assessing the overall quality of the speech coding, we employed an objective test called POLQA and a subjective listening test based on the MUSHRA methodology. In this section we describe briefly each of them.

Spectral Distortion
Mean squared log spectral distortion to which we refer as spectral distortion (SD), is an objective measurement used for we evaluating our different methods for envelope. It is formulated as:

$$SD = \sqrt{\frac{1}{k}\left(\sum_{i=1}^{k}(10log(hr_i) - 10log(ha_i))^2\right)}, \tag{2.8}$$

where ha_i is the real-valued power spectrum of the modelled spectral envelope, hr_t is the reference to which we compare ha_i and k is the length of ha_i and hr_t.

POLQA
POLQA is a voice quality testing standard stands for "Perceptual Objective Listening Quality Assessment".

It compares each sample of the reference signal (talker side) to each corresponding sample of the degraded signal (listener side) [19]. Perceptual differences between both signals are scored as differences. These differences, which are the results of the POLQA test, are in Mean Opinion Score(MOS) scale from 1 (bad) to 5 (excellent).

MUSHRA
MUltiple Stimuli with Hidden Reference and Anchor (MUSHRA) is subjective listening test to evaluate the perceived quality of output from speech and audio codecs [18].

In MUSHRA, the listener is provided with the known reference in one side and a certain number of test samples, a hidden version of reference and an anchor in the other side. Listener should be able to distinguish the hidden reference. Otherwise, the result of the test would not be valid. According to their assessments, listeners grade the samples from 0 to 100. The results are in the MOS scale.

In MUSHRA, the listener is presented with the reference (labeled as such), a certain number of test samples, a hidden version of the reference and one or more anchors. The recommendation specifies that one anchor must be a 3.5 kHz low-pass version of the reference. The purpose of the anchor(s) is to make the scale be closer to an

"absolute scale", making sure that minor artifacts are not rated as having very bad quality. The results of MUSHRA listening test mus be presented by [18]:

- description of the test materials.
- number of listeners.
- the overall mean score for all test items used in the experiment.
- the mean scores and 95% confidence interval of the statistical distribution (e.g. t-distribution)

3 Full Band Envelope Coding using Linear Predictive Coding

Linear Predictive Coding (LPC) is a key component of CELP based speech codecs. Linear prediction analysis is used to represent the spectral envelope of a speech signal [7] and models the vocal tract in the source-filter model of speech production. In the course of the thesis LPC is one of two techniques by which we model the spectral envelope for full-band speech coding. In this chapter we briefly explain LPC analysis along with our investigation for extending it full-band speech coding.

The chapter starts with giving definition of linear prediction before explaining how linear prediction coefficients are estimated. Afterwards, we shortly discuss pre-processing tools used in LPC analysis. After explaining line spectral frequencies, we give LPC configuration generally used for wide-band speech coding. Following this, we present our investigations on LPC and vector quantizer to get an appropriate spectral envelope and vector quantizer for the purpose of full-band speech coding. Finally, we evaluate a full-band speech coding scheme integrating the proposed spectral modeling by both objective and subjective quality tests.

3.1 Definition of Linear Prediction

The basic idea of linear prediction is to estimate/*predict* current sample of the signal by *linear* combination of a defined number of past samples, it can be defined as [25]:

$$\hat{s}(n) = \sum_{i=1}^{M} a_i s(n-i), \tag{3.1}$$

where $\hat{s}_n$ is the estimation of the current sample, M is the number of the past samples, also known as prediction order, a_i, the predictor parameters called as well linear prediction (LP) coefficients. Since Equation (3.1) represents an *estimation* of the current sample, there is an inevitable prediction error which can be defined as:

$$e(n) = s(n) - \hat{s}(n) = s(n) + \sum_{i=1}^{M} a_i s_{n-i}, \tag{3.2}$$

where $e(n)$ is the error signal which is known as residual signal. The objective of such a prediction is to minimize $e(n)$ to get the best estimation of $s(n)$. Therefore, in LPC analysis LP coefficients a_i are optimized such that $e(n)$ is minimized. To do so, one efficient way is to minimize: mean squared error (MSE) $E = \sum_{n=-\infty}^{\infty} e(n)^2$ [25], [7].

There are two approaches to estimate LP coefficients a_i such that the mean squared error is minimized. They are known as Autocorrelation Method and Covariance

Method. Since autocorrelation method is used in CELP codec, we explain it in the next section, for covariance method [25] and [30] can be used.

3.2 Autocorrelation Method

Speech signals are changing continuously and their statistical properties are changing over time, however they change slowly and are usually considered quasi-stationary on 20 ms segments. Accordingly, for analysis purposes, the input signal is split into short enough segments, for which the signal is considered stationary. An LP analysis is then performed for each segment and LP coefficients are computed about every 20 ms.

3.2.1 Windowing

Windowing is the process of splitting a signal into successive short segments [7]. The *windowed signal* $s_w(n)$ is achieved by multiplying a windowing function $w(n)$ to the signal $s(n)$. Since $w(n)$ determines the weight of each sample in the MSE estimator, , choosing the appropriate $w(n)$ is crucially important. One simple windowing is rectangular window [23]:

$$w(n) = \begin{cases} 1; & 0 \leq n \leq N-1, \\ 0; & otherwise. \end{cases} \tag{3.3}$$

where N is the window length. From Equation (3.3) it can be seen that a sudden discontinuity at the window borders. This sudden discontinuity in the time domain results in undesirable large side lobes in the frequency domain [29]. To avoid this side lobe oscillation in frequency domain, we need a windowing function whose borders goes smoothly to zero in time domain. Hamming window causes to overcome the mentioned problem. It is widely used in LPC analysis and defined as [23]:

$$w(n) = \begin{cases} 0.54 - 0.46cos\left(2\pi \frac{n}{N-1}\right); & 0 \leq n \leq N-1, \\ 0; & otherwise. \end{cases} \tag{3.4}$$

In the present work, we adopt Hamming windowing. Figure 3.1 shows the Hamming window with length 64 along with the frequency response. Some other windowing functions, which have been used in different speech codecs, are Hann, Bartlet, Kaiser and Blackman window functions [23].

3.2.2 Estimation of the Autocorrelation

In the autocorrelation method the windowed signal $s_w(n)$ is considered::

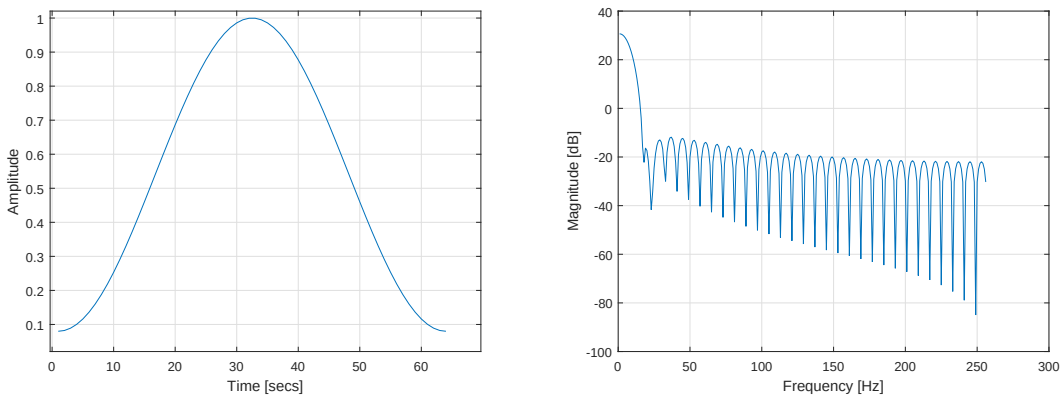


Figure 3.1: Hamming window function and its frequency response.

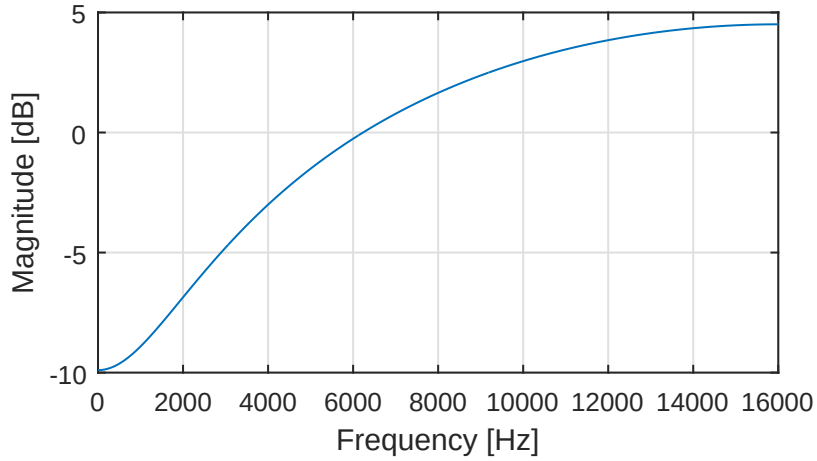


Figure 3.2: Pre-emphasis filter with $\alpha = 0.68$.

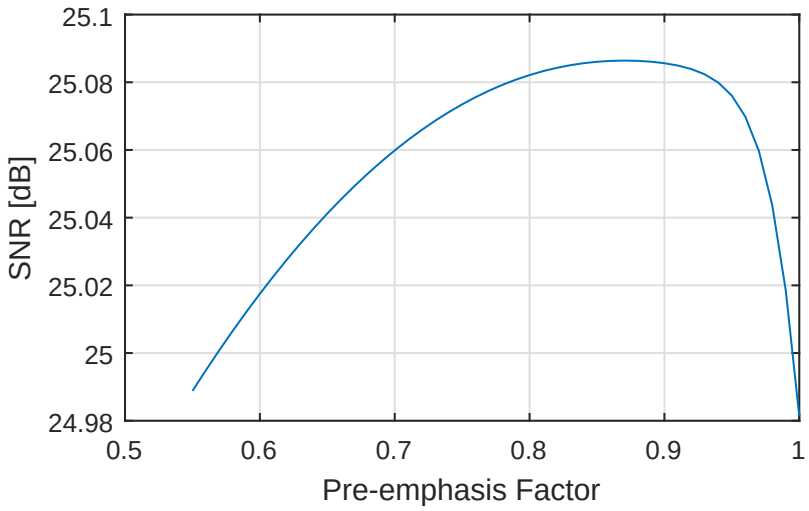


Figure 3.3: SNR with different values of pre-emphasis factor α. Maximum SNR is found for α around 0.85.

$$s_w(n) = w(n)s(n), \tag{3.5}$$

The MSE can then be formulated as follows:

$$E = \sum_{n=-\infty}^{\infty} \left(s_w(n) - \sum_{i=1}^{M} a_i s(n-i) \right)^2, \tag{3.6}$$

For minimizing Equation (3.6) we set $\frac{\partial E}{\partial a_i}$, for $i = 1, ..., M$, consequently M equations with M unknown a_i are obtained:

$$\sum_{j=1}^{M} a_j \sum_{n=-\infty}^{\infty} s_w(n-i)s_w(n-j) = \sum_{n=-\infty}^{\infty} s_w(n-i)s_w(n), 0 \leq i \leq M. \tag{3.7}$$

On the other hand, autocorrelation function of windowed signal $s_w(n)$ can be defined as:

$$R(i) = \sum_{n=-\infty}^{\infty} s_w(n)s_w(|n-i|), \tag{3.8}$$

$R(i)$ is an even function ($R(i) = R(-i)$). Considering Equation (3.8) in Equation (3.7) we obtain:

$$\sum_{j=1}^{M} R(|i-j|)a_j = R(i), 0 \leq i \leq M. \tag{3.9}$$

The matrix form of the equation (3.9) is:

$$\begin{bmatrix} R(0) & R(1) & \cdots & R(M-1) \\ R(1) & R(0) & \cdots & R(M-2) \\ \vdots & \vdots & \ddots & \vdots \\ R(M-1) & R(M-2) & \cdots & R(0) \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_M \end{bmatrix} = \begin{bmatrix} R(1) \\ R(2) \\ \vdots \\ R(M) \end{bmatrix} \tag{3.10}$$

The above matrix is Toeplitz matrix (symmetrical with a constant diagonal). This property of the matrix lets us apply a very efficient recursive procedure "Levinson-Durbin" to obtain LP coefficients a_i. Levinson-Durbin algorithm is depicted in Figure 3.4.

The autocorrelation method using Levinson Durbin algorithm is one of the most common algorithm for estimation of LP coefficients a_i. It is mentioned in most of speech coding books such as [7] and [22].

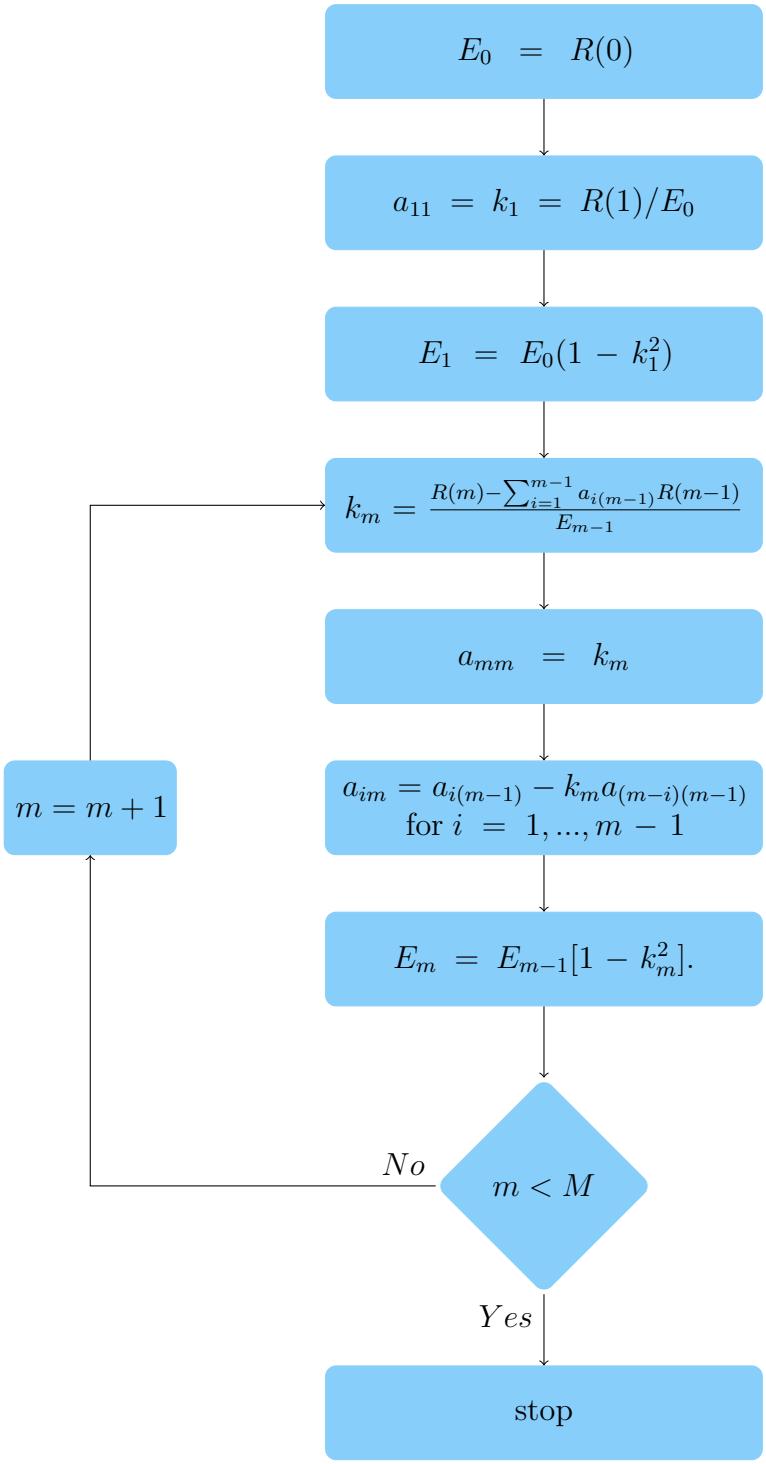


Figure 3.4: Levinson-Durbin recursion algorithm.

3.3 Pre-processing Tools

For improving the LP analysis, different pre-processing steps on the input signal are performed, namely $pre-emphasising$, $lag-windowing$ and $white-noise\ \ correction$.

3.3.1 Pre-emphasising

For LP analysis and in case of sample rate 12.8 kHz and higher, speech signals generally have too high energy in low frequencies. That is, during the speech production procedure high-frequency part are suppressed and the speech signal is dominated by low-frequency components. This property causes ,in LPC analysis, harmonic structure to be modelled [23]. This undesirable phenomenon is overcome by pre-emphasis tool. It compensates *excessive* emphasis on low frequencies and improves the overall tilt of the signal [7].

Pre-emphasis is in general implemented as a first order high pass filter of the input signal is:

$$p(z) = 1 - \alpha z^{-1}, \tag{3.11}$$

where α is a constant which is tuned experimentally. It is known as pre-emphasis factor and in wide-band ,case using a sampling rate of 12.8 kHz for the coding,the pre-emphasis factor is set to 0.68. Figure 3.2 shows the frequency response of a pre-emphasis filter with $\alpha = 0.68$.

3.3.2 Lag-windowing

For high-pitch voiced speech signals, linear prediction has problem to predict the envelope accurately. High-pitch signals have large harmonic spacing which causes linear prediction underestimating the bandwidth of the formants. This problem can be minimized by the procedure called lag-windowing. Lag-windowing helps to stabilize and estimate more accurately LP coefficients [23]. We apply lag-windowing by multiplying the pre-emphasised speech signal to a $lag\ \ window$ function which is usually a Gaussian function. Equation 3.12 is an example of lag-windowing.

$$r_l(i) = e^{-0.5(\frac{2\pi f_0 i}{f_s})^2} r(i), i = 1, ..., M \tag{3.12}$$

where f_0 is bandwidth expansion of the lag window, f_s is the sample rate, $r(i)$ is the autocorrelation of pre-emphasised signal and M is LPC order.

3.3.3 White-noise Correction

In autocorrelation matrix, the eigenvalues might go such low that it leads to large values of LP coefficients. This problem can be alleviated by adding an artificial noise-floor to the signal in order to avoid stability problems when estimating the LP coefficients [5].

3.4 Line Spectral Frequencies

Line Spectral Frequencies (LSFs) are a representation of LP coefficients. In speech codecs, LSFs are usually preferred to direct form of LP coefficients, because in quantization process, direct form of LP coefficients are more sensitive to quantization error, especially when the order is getting high [23].

Moreover, LSFs are bounded to the range of 0 to π , which make easier the design of the quantization. Another superiority of LSFs over direct form is that it can be seen as an interpretation of frequency domain (with multiplication of LSFs by f_s/π LSFs can be converted to frequency range Hz). Since high frequency components are perceptually less important than low frequency components, we can quantize LSFs corresponding to high frequencies with fewer bits. This allows us to allocate fewer bits to LSFs perceptually less relevant, and in that way save bits in coding compared to schemes adopting the direct form of LP coefficients [23].

3.4.1 Computation of Line Spectral Frequencies

Linear prediction can be an all-zero filter $A(z) = 1 + \sum_{i=1}^{M} a_i z^{-i}$, M is the order of the filter and a_i refers to LP coefficients. We can also express $A(z)$ as:

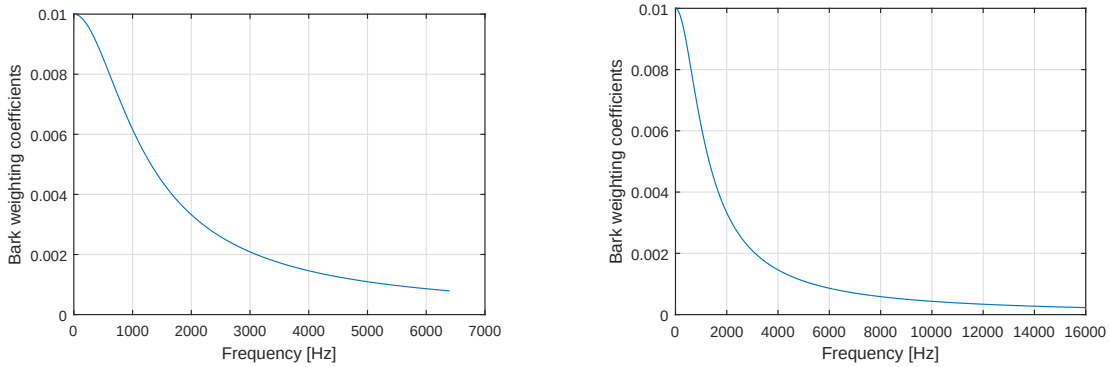
$$A(z) = \frac{1}{2}\left(P(z) + Q(z)\right), \tag{3.13}$$

with:

$$P(z) = A(z) + z^{-(p+1)}A(z^{-1}), \tag{3.14}$$

$$Q(z) = A(z) - z^{-(p+1)}A(z^{-1}), \tag{3.15}$$

When $A(z)$ is minimum phase, one important property of $P(z)$ and $Q(z)$ is that all roots of $P(z)$ and $Q(z)$ are on the unit circle and the roots of $P(z)$ alternate with those of $Q(z)$ [34]. The angular positions of these roots are LSFs with $0 \leq \omega_i \leq \pi$. To find the roots of $P(z)$ and $Q(z)$ (LSFs) several solutions such as applying a discrete cosine transformation [35] and using Chebychev polynomials [21] have been proposed.



(a) Wide-band, sample rate 12.8 kHz.

(b) Full-band, sample rate 32 kHz.

Figure 3.5: Weights derived from bark scale used in wide-band and full-band cases.

3.5 Weighting

Human's ear is more sensitive to low frequency components than high frequency components. Weighting is considering this uneven sensitivity in quantization process such that LSFs corresponding to low frequency components are weighted more than LSFs corresponding to high frequencies. In wide-band speech coding weighting is based on defining the spectral sensitivity regarded to LSFs [14] along with additional Bark scaled coefficients which coordinate with the response of human's ear to a sound [26]. Bark weighting can be defined as:

$$w(f) = \frac{1}{25 + 75\left(1 + 1.4(\frac{f}{1000})^2\right)^{0.69}}, \tag{3.16}$$

where f is frequency from 0 to nyquist frequency. The coefficients of an all-pole filter with this response are determined and stored in advance as the perceptual weighting filter [26]. Figure 3.5 depicts the weights derived from equation 3.16 with corresponding frequencies in wide-band and full-band cases.

3.6 Optimal LPC in Wide-Band Envelope Modelling

LPC envelope modelling has been used in wide-band speech coding standards such as EVS [2] and AMR-WB, [1]. From the numerous previous works, one can derived the different optimal parameters for LPC for WB speech. The key parameters of such an optimal coding are listed below:

- Sampling rate f_s: 12.8 kHz
- Gamma factor γ_1: 0.92 (the weighting coefficient used in perceptual model)
- Pre-emphasis factor α: 0.68

- LPC order M: 16

For quantization, the technique moving average multistage vector quantization (MA-MSVQ) was shown to be one of the most efficient quantization scheme for teh LSFs. The configuration of MA-MSVQ typically used fro WB is:

- Number of bits: 31

- Constellation: [7 6 6 6 6]

- M in $M-best$ search: 8

- α prediction constant for MA prediction: 0.33

Also weighting technique explained in 3.5 is applied on LSFs in quantization procedure.

3.7 LPC in Full-Band Envelope Modelling

As we mentioned in chapter 1, we extend LPC envelope modelling used in wide-band to meet needs in full-band speech coding. To do so, we must derive new values for the parameters explained in 3.6. Sampling rate for full-band speech coding is 48 kHz [10] but we consider in this work the case where the input signal is sampled at 32kHz, historically considered as SWB case, although it can be easily extended to signals sampled at 48 kHz, i.e. FB case. Moreover, if our implentation with sampling rate 32 kHz works appropriately, it can be adapted to 48 kHz. The weighting coefficient $\gamma 1$ is set to 0.92. The estimation of LPC order M and pre-emphasis factor α is explained in the following sections.

3.7.1 Estimation of LPC Order

For estimating LPC order M, we use measurement: signal to noise ratio (SNR) in which signal is input signal before pre-emphasising and noise is excitation signal. First we measure SNR in wide-band to investigate the behavior of LPC with different values for LPC order from 12 to 20. Results show that after order 16, which is the optimal LPC order in wide-band, SNR increment per additional order becomes less than 0.1 dB. Therefore, we target to have SNR increment less than 0.1 dB per additional order in full-band. With the same measurement of SNR with LPC order from 16 to 32, the first 0.1 dB SNR increment occurs LPC order 22. Therefore, we can consider order 22 a candidate for full-band. But since our quantizer is multi stage vector quantizer, we choose LPC order 24 (3*8) which is more appropriate value for designing the quantizer (LPC $M = 24$). Table 3.1 shows the results of SNR measurement for both wide-band and full-band.

Table 3.1: SNR of wide-band and full-band LPC for order estimation in full-band.

Wide-Band Order	12	13	14	15	16	17	18	19	20
SNR [dB]	22.2084	22.3854	22.5295	22.6481	22.7634	22.8592	22.9589	23.0428	23.1328

Full-Band Order	18	19	20	21	22	23	24	25	26
SNR [dB]	31.7400	31.8535	31.9843	32.0855	32.2096	32.3052	32.4024	32.5016	32.6006

3.7.2 Estimation of Pre-emphasis Factor

For estimation of the pre-emphasis factor we also use SNR measurement. we change the pre-emphasis factor form 0.5 to 1 with pace 0.2 and calculate the SNR for each case. Figure 3.3 shows that in pre-emphasis factor 0.85 SNR is maximized. Therefore we choose this value for pre-emphasis factor ($\alpha = 0.85$).

3.7.3 Training Codebook

As we mentioned in chapter 1 section 2.4, we use MA-MSVQ to quantize LSFs. therefore, we need to train a codebook for quantization. To do so, we must take into account the following parameters:

- number of bits.
- bit structure (constellation of the bits/number of stages).
- number of chosen best values for searching in each stage($M - best$).

For performance analysis of the quantizer we have metrics:

- average SD alongwith 2-4 dB outliers, $>$ 4 dB outliers.
- Bark weighted average SD alongwith 2-4 dB outliers, $>$ 4 dB outliers.

, where we compare quantized LSFs to unquantized LSFs. It is usually considered that a good quality is achieved if the following conditions are fulfilled [28]:

- average SD less than 1 dB.
- 2-4 dB outliers $<$ 2%.
- no 4 dB or more outliers.

However, to get the best overall performance with given number of bits, we consider the trade off between average SD and the outliers (larger SD in return of fewer outliers).

Since training the codebook is an offline procedure and is carried out once, we do not consider complexity of the training process.

3.7.3.1 Codebook in Wide-Band

For wide-band, with configuration of the MSVQ in 3.6 we get results presented in table 3.2.

Table 3.2: SD results of codebook training in wide-band.

M-best = 8	Average SD [dB]	Outlier 2−4 dB[%]	Outlier > 4 dB[%]
31 bits: [7 6 6 6 6]	1.31	2.04	0.00

M-best = 8	Bark weighted Average SD [dB]	Outlier 2−4 dB[%]	Outlier > 4 dB[%]
31 bits: [7 6 6 6 6]	1.57	20.94	0.30

3.7.3.2 Codebook in Full-Band

The objective of the experiment is to achieve as accurate quantizer as in wide-band with as few bits as possible. Since possibilities for number of bits and the related constellation can be very large, we limit the experiment environment to:

- number of bits from 35 to 39 bits.
- number of stages: 5 or 6.
- each structure does not have more than 1 stage with 9 bits and 2 stages with 8 bits.
- value of M-best from 6 to 12 with pace 2.

We experimented 136 possibilities with above-mentioned combinations. Table 3.3 shows the best result for each number of bits. We can see that vector quantizer with 39 bits [9 8 8 7 7] gives the best result. Therefore we choose this configuration for our vector quantizer.

Table 3.3: SD results of codebook training in full-band.

M-best = 8	Average SD [dB]	Outlier 2−4 dB[%]	Outlier > 4 dB[%]
35 bits: [9 7 7 6 6]	1.52	7.37	0.27
36 bits: [9 7 7 7 6]	1.48	5.81	0.14
37 bits: [9 8 7 7 6]	1.43	4.29	0.00
38 bits: [9 8 7 7 7]	1.39	3.35	0.00
39 bits: [9 8 8 7 7]	1.35	2.32	0.00

M-best = 8	Bark weighted Average SD [dB]	Outlier 2−4 dB[%]	Outlier > 4 dB[%]
35 bits: [9 7 7 6 6]	1.98	38.31	3.54
36 bits: [9 7 7 7 6]	1.93	36.43	3.17
37 bits: [9 8 7 7 6]	1.88	34.80	2.77
38 bits: [9 8 7 7 7]	1.84	33.45	2.41
39 bits: [9 8 8 7 7]	1.78	30.69	2.28

Table 3.4: Different SD measurements of obtained LPC envelope and quantizer.

	SD	Outliers 2−4 dB[%]	Outliers > 4 dB[%]
unqantized LPC	3.1678	81.99	13.41
quantized LPC	3.5071	71.06	25.74

(a) SD with reference true envelope.

	weighted SD	Outliers 2−4 dB[%]	Outliers > 4 dB[%]
unqantized LPC	3.9889	55.15	37.58
quantized LPC	4.6221	34.22	60.05

(b) Bark weighted SD with reference true envelope.

	SD	Outliers 2−4 dB[%]	Outliers > 4 dB[%]
quantized LPC	1.7335	12.72	0.04

(c) SD with reference unquantized LPC.

	weighted SD	Outliers 2−4 dB[%]	Outliers > 4 dB[%]
quantized LPC	2.2951	48.98	6.88

(d) Bark weighted SD with reference unquantized LPC.

3.8 Experiments and Results

We evaluate the obtained LPC envelope and vector quantizer by objective measurements SD and POLQA and subjective evaluation MUSHRA listening test.

3.8.1 SD

We use SD to have an assessment of LPC spectral envelope model and vector quantizer independent from the speech codec. Table 3.4 shows the results of different SD measurements. In table 3.4a we can see that SD of quantized LPC envelope, when compared to the true envelope as defined in section 2.2, is around 0.4 dB worse than LPC envelope without quantization. This difference is reasonable as the quantization is inevitably accompanied by loss in SD. Table 3.4b is weighted SD with reference true envelope. Weighting SD is based on equation (3.16). From the table we can see deterioration in SD compared to result in table 3.4a. This results from the fact that we have not applied any weighting on LSFs. That is why, later we adapt weighting to full-band case. Also, the SD and weighted SD of quantized LPC envelope model compared to LPC without quantization are presented respectively in table 3.4c and 3.4d.

3.8.2 MUSHRA Listening Test

To have the overall quality assessment of the speech coding we use MUSHRA listening test methodology. For this test we evaluated 12 speech items consisting of 6 noisy speech items and 6 clean speech items in German, English and French languages, 8 listeners including 6 expert listeners took part of the test. We analyse the results of MUSHRA test by using student's t-distribution with 95% confidence interval.

Listeners were required to assess conditions: hidden reference, anchor, ACELP+SBR (ACELP_SBR) which is xHE-AAC audio and speech codec [12], FB-CELP LPC (FB_CELP) which refers to our LPC envelope modelling and TCX+SBR (TCX) hybrid codec [8].

Figure 3.6 and Figure 3.7 depict the average absolute MUSHRA scores for respectively clean speech and nosiy speech items. We can see from the figures that for all items ACELP_SBR has better quality than FB_CELP. Especially, the quality difference between ACELP_SBR and FB_CELP is more audible in clean speech items. However, in noisy speech FB_CELP gets closer to ACELP_SBR and even for item $german\quad female\quad office$ FB_CELP performs as good as ACELP_SBR. For all items, FB_CELP generally performs worse than ACELP_SBR and statistical analysis using student's t-distribution confirms our observation. To sum up the observations, the extension of LPC and quantizer for full-band speech coding was successfully integrated in a complete full-band speech coding scheme. However, we need to improve our implementation. To do so, first we have to know this degradation in quality results from LPC envelope model or vector quantizer. Therefore, we set up another MUSHRA test with same environment but this time listeners were asked to evaluate: hidden reference, anchor, LPC24_Q which is quantized LPC with order 24 (our current configuration) , LPC24_UQ which is unquantized LPC with order 24 (to see effect of quantizer), LPC32_UQ which is unquantized LPC with order 32 (to see effect of LPC).

Figure 3.8 shows that both LPC envelope model and vector quantizer cause the quality to drop. So, we need to improve quantizer and LPC envelope model. In the following sections we explain how we improve our implementation by applying weighting on LSFs quantization and by amending the perceptual model derived from the LP coefficients.

3.9 Weighting in Full-Band

Weighting technique explained in section 3.5 has shown to improve the perceptually weighted SD and the perceived quality in wide-band case [2]. Therefore, we aim to extend this weighting technique to full-band case. To do so, the weighting coefficients, which are based on bark sale, must be redefined according to the sample rate used in full-band envelope modelling. The results of applying weighting on LSFs are shown in table 3.5. From the results we can see significant improvement on the

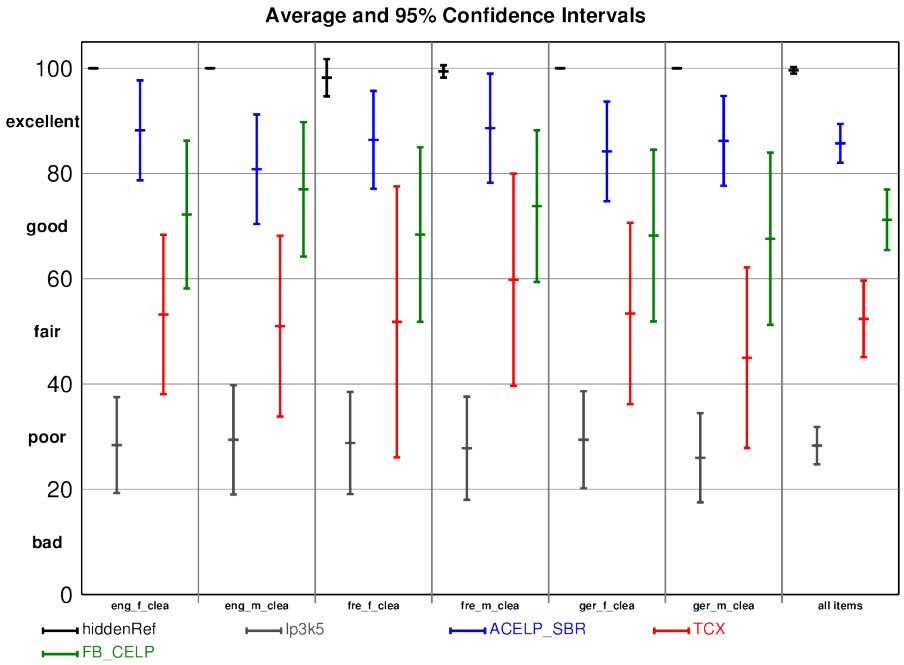


Figure 3.6: Average absolute MUSHRA scores for 6 clean speech items using 95% confidence intervals of t-distribution.

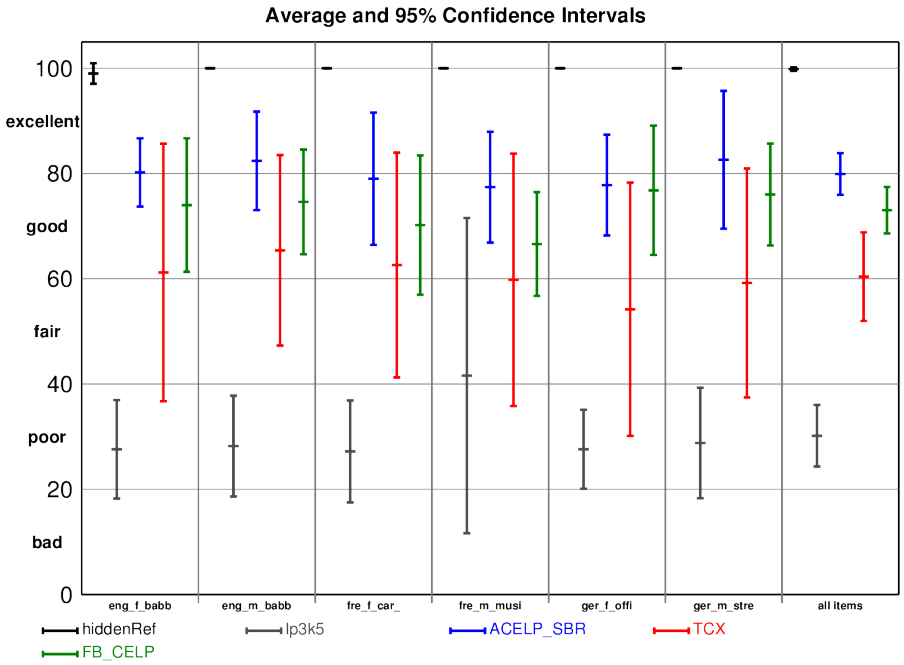


Figure 3.7: Average absolute MUSHRA scores for 6 noisy speech items using 95% confidence intervals of t-distribution.

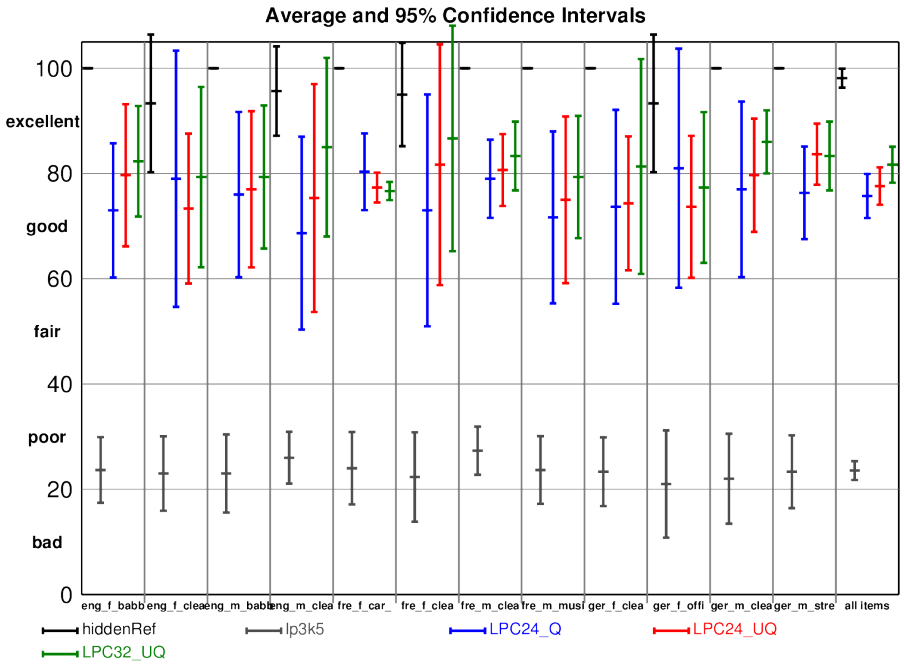


Figure 3.8: Average absolute MUSHRA scores for different configuration of our implementation.

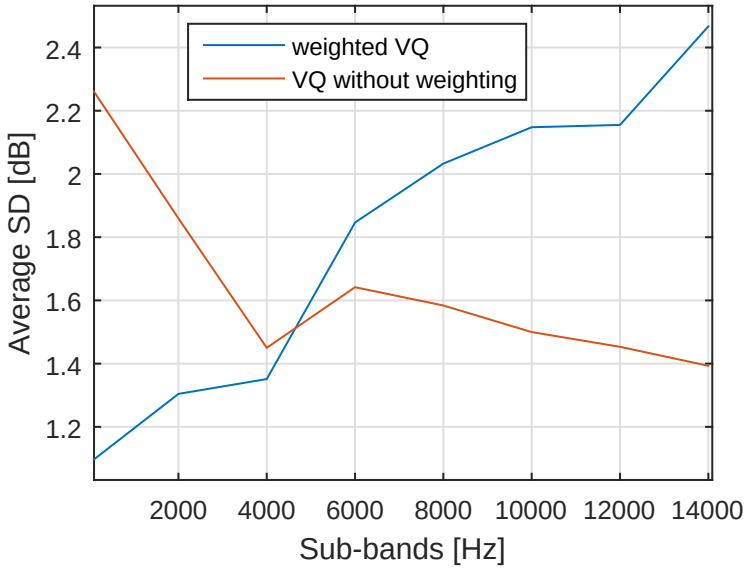


Figure 3.9: Sub-band analysis of measurement shown in the table 3.5b.

Table 3.5: SD measurements of quantized LPC envelope with and without weighting.

	SD	Outliers 2−4 dB[%]	Outliers > 4 dB[%]
quantized LPC	3.5071	71.06	25.74
weighted quantized LPC	3.5930	72.87	26.26

(a) SD with reference true envelope.

	SD	Outliers 2−4 dB[%]	Outliers > 4 dB[%]
quantized LPC	1.7335	12.72	00.04
weighted quantized LPC	1.9805	43.25	00.17

(b) SD with reference unquantized LPC.

	weighted SD	Outliers 2−4 dB[%]	Outliers > 4 dB[%]
quantized LPC	4.6221	34.22	60.05
weighted quantized LPC	4.1367	53.71	41.21

(c) Bark weighted SD with reference true envelope.

	weighted SD	Outliers 2−4 dB[%]	Outliers > 4 dB[%]
quantized LPC	2.2951	48.98	6.88
weighted quantized LPC	1.1802	8.04	0.28

(d) Bark weighted SD with reference unquantized LPC.

bark weighted average SD. Moreover, to investigate the impact of weighting on low frequency components, we split the bandwidth into 8 sub bands and measure the average SD for each sub-band. This measurement shows that weighting leads to a significantly lower SD for low frequency components. As an example, Figure 3.9 is the sub-band decomposition of the SD given in table 3.5b. We can observe that even though average SD of weighted LPC is around 0.25 dB worse than LPC without weighting, weighting improves SD of low frequency components which are perceptually more relevant than high frequency components.

3.10 Estimation of Perceptual Model Parameters

In section 3.7 we used SNR to estimate LPC order and pre-emphasis factor. However, MUSHRA listening test 3.8 showed that by increasing LPC order LPC, the overall quality is improved. Hence, we reconsider estimation of the LPC parameters: LPC order, pre-emphasis factor and gamma factor. To do so, we design an experiment in which we find the optimal values for mentioned parameters by minimizing the SD in band 0-6.4 kHz (nyquist frequency in wide-band). between the new full-band LPC scheme and the conventional wide-band LPC scheme. Figure 3.10 shows that the optimal values for LPC order, pre-emphasis factor and gamma factor are found to be:

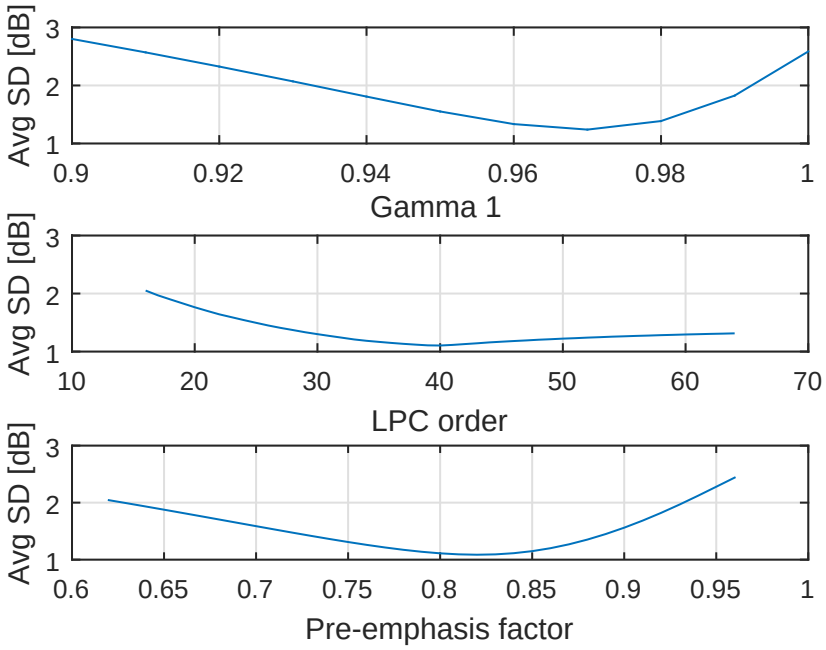


Figure 3.10: Estimation of LPC order M, pre-emphasis factor α and gamma factor $\gamma 1$ used in perceptual (psychoacoustic) model.

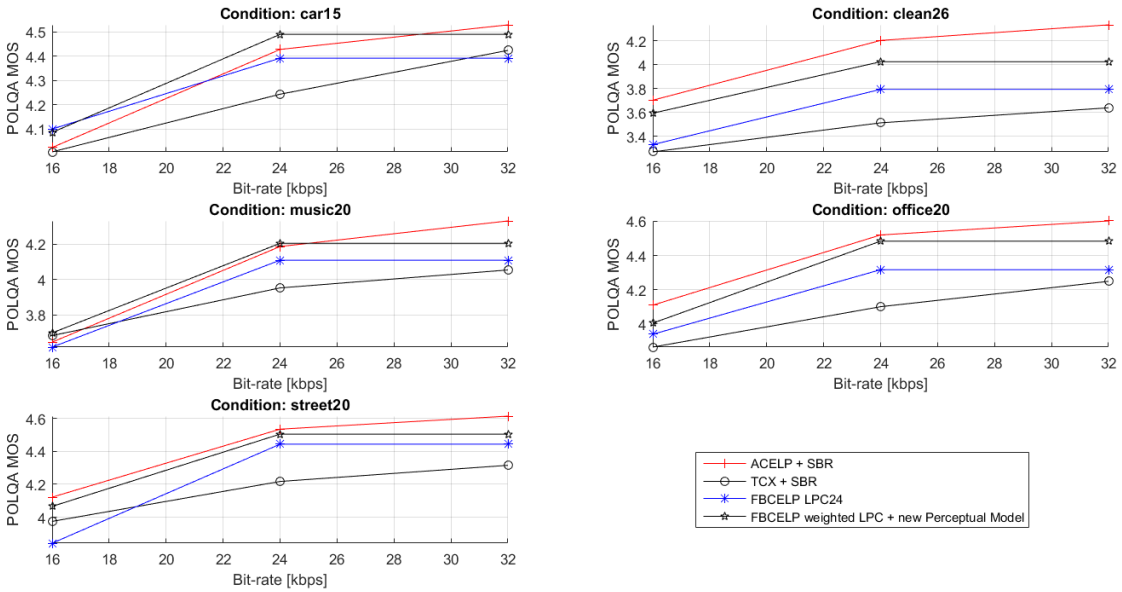


Figure 3.11: POLQA measurement for evaluation of weighting and new perceptual model.

- LPC order $M = 40$.
- pre-emphasis factor $\alpha = 0.82$.
- gamma factor $\gamma 1 = 0.97$.

We use these values in the perceptual model of the speech codec. The reason why we do not use these parameters in LPC envelope model is that in vector quantizer, LPC order 40 might results in very large complexity which could be impossible to implement in practical application.

We use POLQA test to evaluate the effect of weighting and new perceptual model on overall quality of the speech coding. Figure 3.11 shows the result of POLQA measurement. It compares ACELP_SBR (red line), TCX_SBR (black line with circle marker), FB_CELPLPC24 our previous quantization design (blue line) and FB_CELPLPC24 with weighting and new perceptual model (black line with star marker). From the figure we can see that weighting along with new perceptual model results in great improvement on the quality of the speech. For all speech items, it has significantly better performance than our previous LPC envelope model and for noisy items it is either as good as ACELP_SBR or very close to ACELP_SBR. To confirm the result derived from POLQA test we conducted another MUSHRA listening test. The results of MUSHRA listening test are given and analyzed in the chapter 5.

4 Full Band Envelope Coding using Distribution Quantization

Distribution quantization (DQ) is a technique for modeling the spectral envelope of a signal. It is based on distribution of the spectral mass of a signal [20]. Currently, modern speech and audio codecs are generally hybrid codecs having separate coding techniques for speech signals and generic audio signals. Audio codecs are usually based on scale factor bands and speech codecs are based on linear predictive coding [24]. Linear predictive coding is a highly efficient technique with cost of complexity, while scale factor representation is a technique with low complexity but not as efficient as linear predictive coding. The final purpose of DQ is to provide a unified codec for both audio and speech signal with high efficiency and low complexity [24].

In this chapter we explain some details regarding DQ and methods by which we can achieve DQ envelope modeling. The chapter starts with the background of DQ and what investigations has been previously launched related to DQ. Afterwards, we describe our contribution on DQ and how we adapt DQ to get the envelope for full-band speech coding. Finally we discuss the results of the experiments regarding to DQ.

4.1 Background of the work

As mentioned before the main idea of DQ is to model a smooth envelope of the spectrum. To find such an envelope, the overall tilt of the spectral envelope is considered. To describe the over tilt different approaches have been examined. One is based on equal magnitude of the spectral mass [20] and the other on is based energy ratio [24]. For both approaches we need certain number of frequencies in the spectrum to compute the arbitrary parameters. To do so, the spectrum of a signal is split into sub-segments with known border frequencies where we do the computations. These frequency borders are known as split points and will be referred with this name in this documentation. In the following section we explain both techniques to position the split points.

In the following sections, we explain two approaches by through which DQ spectral envelope is derived.

4.1.1 DQ Envelope based on Segments with Equal Magnitude

To describe the tilt of the spectral envelope, we can find such frequencies that split the spectrum into segments of equal mass [20]. For this attitude we need cumulative spectral mass which can be mathematically defined as [20]:

$$C_k = \sum_{i=0}^{k} X_i, \tag{4.1}$$

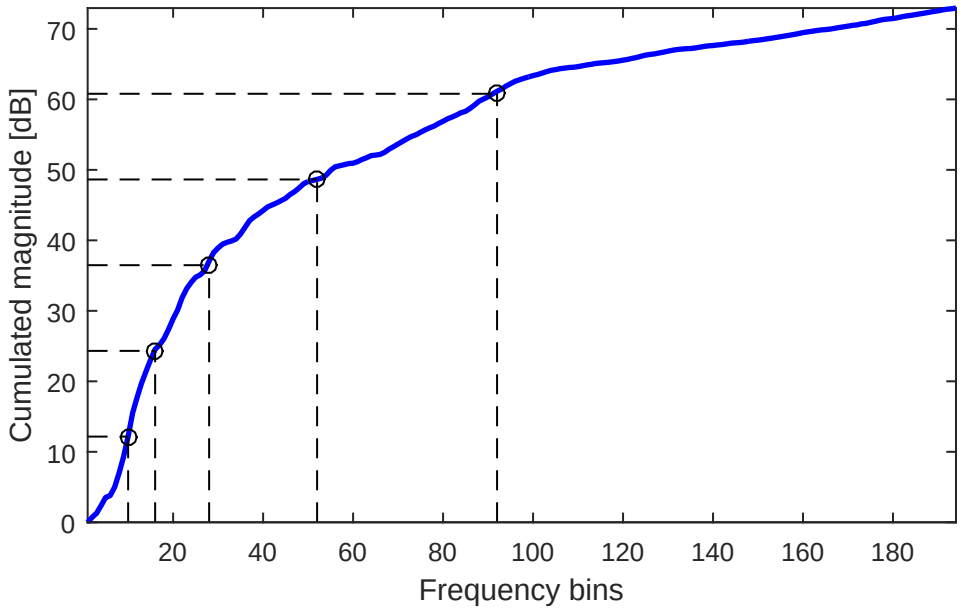


Figure 4.1: Example of frequency bins determination for split points based on equal cumulative spectral mass [20].

where k is the frequency bin upto which the sum of spectral mass is computed, X_i is the magnitude spectrum of the signal at frequency bin m. This computation is done through a windowed signal in frequency domain. With using equation, we are patently able to take total spectral mass of each frame by having k equal to the frequency bin N of Nyquist frequency $C_N = \sum_{i=0}^{N} X_i$.

After computing the total spectral mass, the signal is partitioned into $M + 1$ equal magnitude segments. The frequency bins corresponding to the M segment boundaries are the frequency bins of the split points. Thus, the position of the split points are determined. Figure 4.1 is an example of cumulative spectral mass which is split into 6 segments $M = 5$. The advantage of this method is that as split points have the equal magnitude, for reconstruction of the spectral envelope in the synthesis side, only total spectral mass and frequency bins of the split points are needed. In other words, the magnitude of the each split points can be easily obtained with having the split points and the total spectral mass $C_k/(M + 1)$.
The cumulative spectral mass of m_{th} split point can be obtained from [11]:

$$\hat{C}_k m = m * C_N/(M + 1). \tag{4.2}$$

The next step is to get the cumulative spectral mass of all frequency bins of the frame. This is done by applying interpolation through the split points. Spline interpolation have been shown to give better accuracy for this purpose compared to

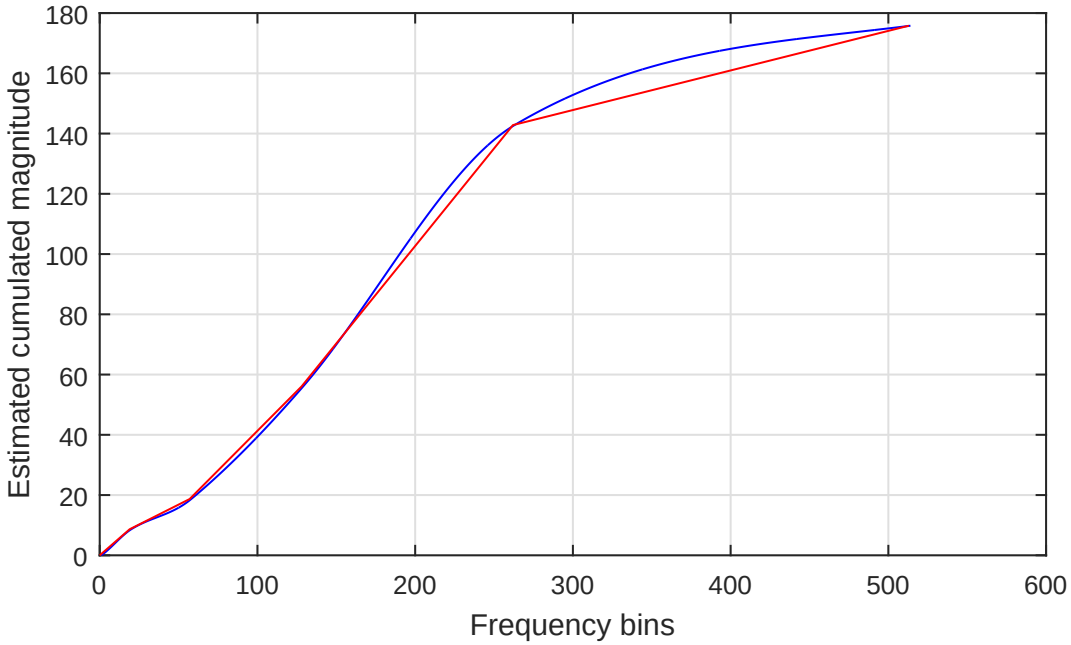


Figure 4.2: Spline interpolation versus linear distribution in estimated cumulative spectral mass [20].

linear interpolation [20]. In spline interpolation, the tilt between the neighboring split points is defined as [20]:

$$T_m = \frac{C_{m+1} - C_{m-1}}{F_{m+1} - F_{m-1}}, \tag{4.3}$$

where T_m is the tilt, m is the split point $1 \leqslant m \leqslant M$, C_m is the cumulative spectral mass of the corresponding split point and F_m is the corresponding frequency bin of the split point. Figure 4.2 shows the difference between spline and linear interpolation.

For the final step, it's the time to get the spectral envelope. With differentiation of the cumulative curve, which is obtained by interpolation, we are doing the transformation into the frequency domain and the spectral envelope is acquired. In Figure 4.3 the envelope results from DQ with spline interpolation and constant values for magnitude is depicted.

Experiments shows that, in terms of entropy (bit consumption), DQ has better result in comparison with scale factor bands and as good as linear prediction. DQ also gives better correlation with the input signal compared to scale factor bands and linear prediction, which consequently causes to have higher SNR [20]. Figure 4.4 is the block diagram of the summary of process done to get the envelope of DQ based on equal cumulative spectral mass C represents *cumulative spectral mass* and SP is for *split point*.

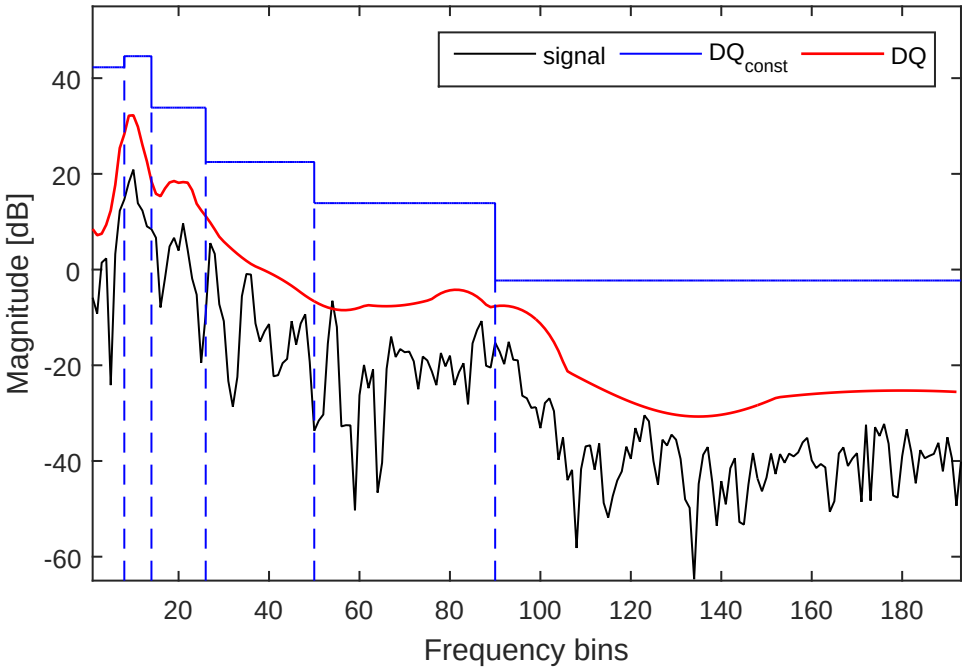


Figure 4.3: In this example the distribution quantizer (DQ) has $M = 5$ splits points, which separate the spectrum into $M + 1$ segments of equal spectral mass. Spline interpolation in cumulative domain gives a smoother envelope in frequency domain (DQ_{int}). For better visibility both envelopes are shifted vertically [20].

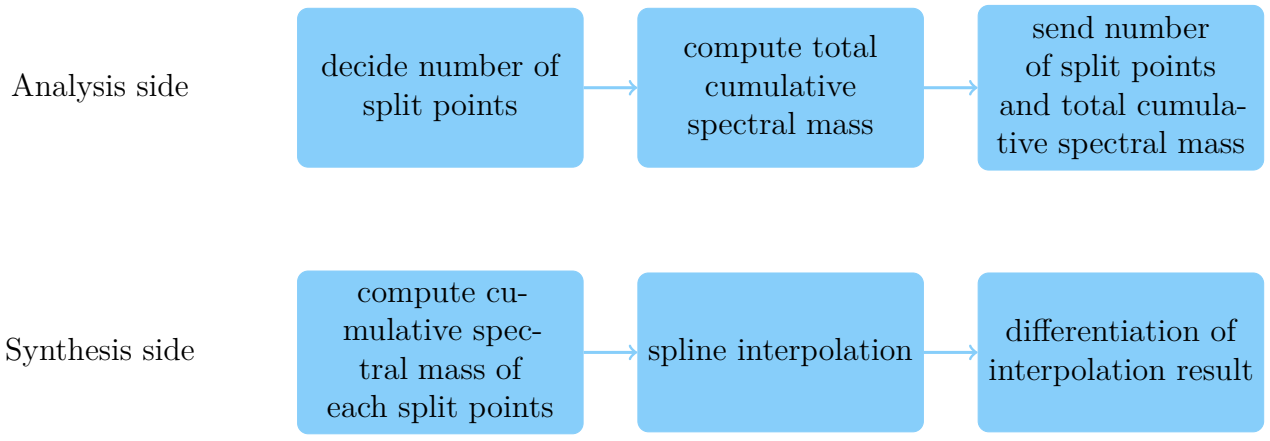


Figure 4.4: Block diagram of analysis and synthesis side for 4.1.1 approach.

4.1.2 DQ used in Entropy Coding for Speech and Audio

Instead of calculating cumulative spectral mass at each split point, normalized cumulative spectral mass (energy ratio) at each split point is computed to describe the overall tilt of the envelope [24]. Also locating the split points is based another technique rather than equal magnitude. Having used a statistical model to quantize the distribution of the spectral mass, an entropy coder is proposed to encode each parameter (energy ratio). To see the performance of proposed technique, it is compared with LPC.

Positioning the split points

As a first step positions of split points are established. For this method in addition to requiring the split points, left and right boundary is also needed. So the establishment split points and and their band width is a level wise structure following tree structure. The logic behind finding boundaries is that in the first level, where we have one split point k_1, left boundary is the beginning of the framebin_1 and the right boundary is frequency bin corresponding nyquist frequency bin_{nyquist}. For the next level, where we have two split points, the boundaries of the split points depend on the split point in the first level. This process will be done recursively up to the depth (number of split points) we desire. Figure 4.5 shows the logic behind of points positioning.

Now the question is how the frequency bins corresponding to the split points are decided. This is done through computation of the signal variance. That is, in the first level, the signal variance of each frequency bin throughout the bandwidth (from 0 to nyquist frequency) is calculated, the frequency bin wherein the maximum variance of the signal happens is considered the first split point. With determination of the first split point the boundaries of the split points of the second level are known (Figure 4.5) for frequency bin throughout of each sub-band, variance of the signal is computed and the bin corresponding to the maximum variance will be location of the split point. This process is recursively done for the next levels and segments.

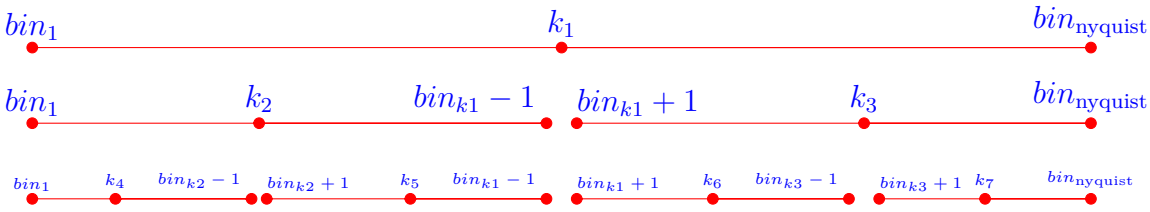


Figure 4.5: Frequency bin corresponding to right and left boundaries of split points with number of split points 7 and number of levels 3.

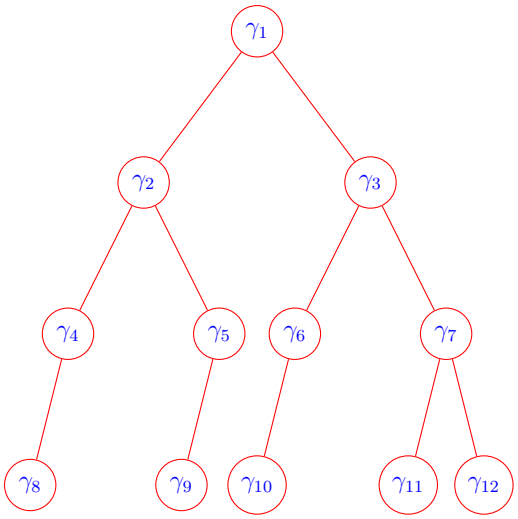


Figure 4.6: Tree structure for number of split points $M = 12$ and level 4.

Once the split points have been chosen, the energy ratios are defined as:

$$\gamma_{ki} = \frac{\sum_{i=left_{ki}}^{k_i} X_i}{\sum_{i=left_{ki}}^{right_{ki}} X_i}, \tag{4.4}$$

where k_i is i_{th} split point, γ_{ki} is the energy ratio of the k_i, $left_{ki}$ is the beginning of the k_i sub-band and $right_{ki}$ is the last frequency bin of the sub-band corresponding to k_i. As an example, using Equation (4.4) for the first split point, the energy ratio will be: $\gamma_1 = \frac{\sum_{i=1}^{k_1} X_i}{\sum_{i=1}^{N} X_i}$, where N is the bin corresponding to nyquist frequency.

With computation of γ of each split it can be realized that γ of each split point depends on the energy ratio of the split point on the upper level. Therefore γ_1 has the impact on all other energy ratios and the impact the energy ratios go down from one level to the next one. Consequently, the accuracy of the quantization should be highest for the γ_1 and having descending approach from a level to the next one. Figure 4.6 shows the distribution of the split points and corresponding energy ratios for number of split points and level, 12 and 4 respectively. From the Figure 4.6 we can see for example γ_8 has dependency on γ_4 , γ_2 and γ_1. Thus, for this example the accuracy of the quantization should be: $accuracy_{\gamma_1} > accuracy_{\gamma_2} > accuracy_{\gamma_4} > accuracy_{\gamma_8}$

Quantization and Coding of Energy Ratios

The recursive structure of obtaining energy ratios results in in-dependency of each γ_k from other γ_k's. In other words, since the bandwidth of each split points is non-overlapping from its siblings (split points in the same level) and also since γ_k of each split point is the *normalized* energy of the left side of the split point, it is independent from γ_k of its parent split point (split point in the upper level). Hence, γ_ks are uncorrelated and can be quantized and encoded independently [24]. This property, allows us to use quantization technique with low complexity. Mid rise

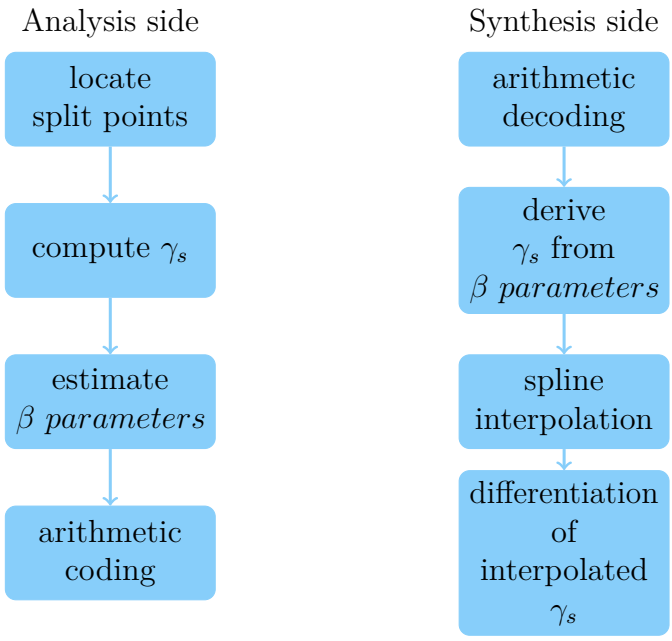


Figure 4.7: Block diagram of analysis and synthesis side for 4.1.2 approach.

(uniform) quantization is one of the quantization technique with low complexity and is used in [24].

To encode energy ratios (γ_ks) efficiently, a statistical model (probability distribution) of each energy ratio should be determined. In [24], Beta-distribution is chosen as a statistical model. Detail explanation of Beta-distribution used for DQ can be found in [11]. In the final step each energy ratio -with using Beta distribution- is encoded through arithmetic coder. In the synthesis side the process is similar to 4.1.1. That is, having decoded the parameters and taking the γ_ks, spline is applied on γ_ks and then with differentiation on γ_ks envelope is resulted.

To evaluate the performance of DQ, SNR as an objective metrics and MUSHRA test as the subjective metrics are considered. Experiments show that in both subjective and objective evaluation, DQ has equal or better performance than LPC with obviously less complexity [24]. Figure 4.7 is the block diagram of the summary of processes done in 4.1.2.

4.2 DQ in Full Band Envelope Modelling

In the course of this thesis, we examine DQ's capability in the use case of super-wideband speech envelope modeling. The way we are using DQ is very similar to 4.1.2 where in it was shown that DQ could be an alternative for hybrid codecs. In our work, in contrast with 4.1.2, we are using fixed split points which are same for all frames and do not depend on the content of signal. We are also trying to overcome obstacles which appear in transition from Wideband to Super-wideband.

4.2.1 Determination of Split Points Positions

As mentioned before we use fixed frequency bins where in split points are located. In order to have psychoacoustically relevant envelope, we decided to have auditory-based positioning structure for split points. With this presumption we chose mel scale which is based on perceptual evaluation.

Mel Scale
The Mel scale is a fundamental result of psychoacoustics, relating real frequency to perceived frequency [36]. In other words, mel scale simulates the logarithmic perception of pitch judged by listeners. There is no single formulation to convert mel to frequency and vice versa. We use equations presented in wikipedia:

$$m = 2595 \log(1 + \frac{f}{700}) = 1127 \ln(1 + \frac{f}{700}), \tag{4.5}$$

where m is a mel point corresponding to the frequency f

$$f = 700(10^{m/2595} - 1) = 700(\mathrm{e}^{m/1127} - 1), \tag{4.6}$$

where f is a mel point corresponding to the frequency m

Figure 4.8 shows frequencies and their corresponding mel scale points for sampling rate 32 kHz. It represents graphically the equations above which are used for sampling rate 32 kHz. From the figure it can be observed that there are more precision in lower frequency than high frequency. This precision make the scale give us more components in low frequencies in comparison with high frequencies.

Mel scale in our case
To apply mel scale to our case we need to fit it to the frame through which we are doing all processes. To do so, first, mel representation of frequency 1 ($mel \quad lower \quad limit$) and the sampling rate ($mel \quad upper \quad limit$) is derived from Equation (4.5). Then with taking $linespace$ between $mel \quad lower \quad limit$ and $mel \quad upper \quad limit$, M points which are linearly spaced in Mel domain are obtained. Finally, for each point m we go back to frequency domain using Equation (4.6). Hence, in frequency domain

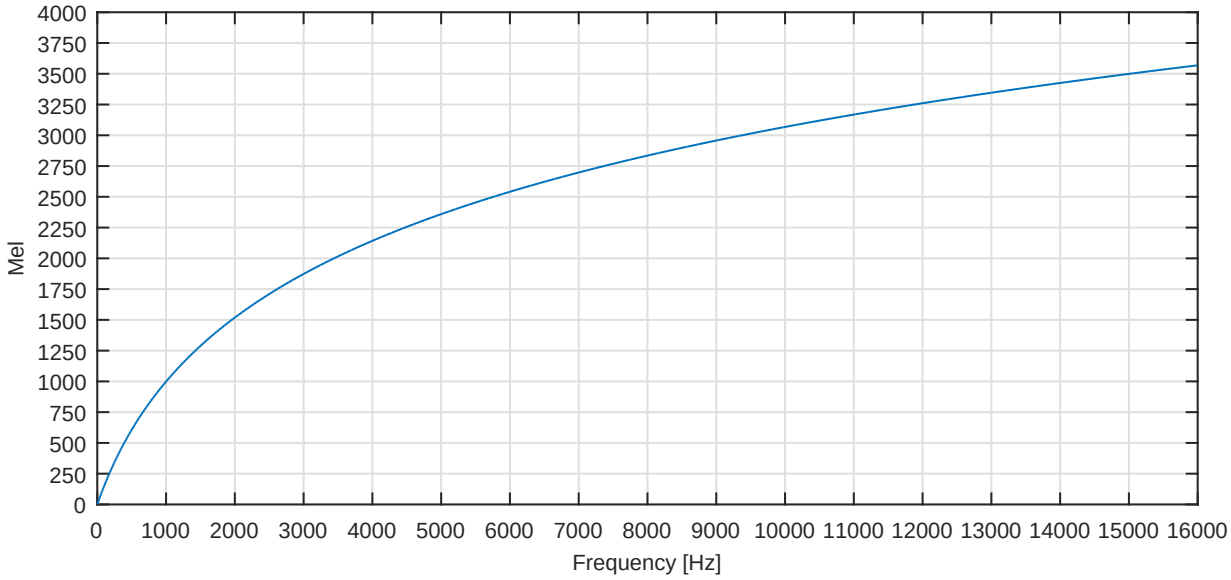


Figure 4.8: Frequency to mel conversion

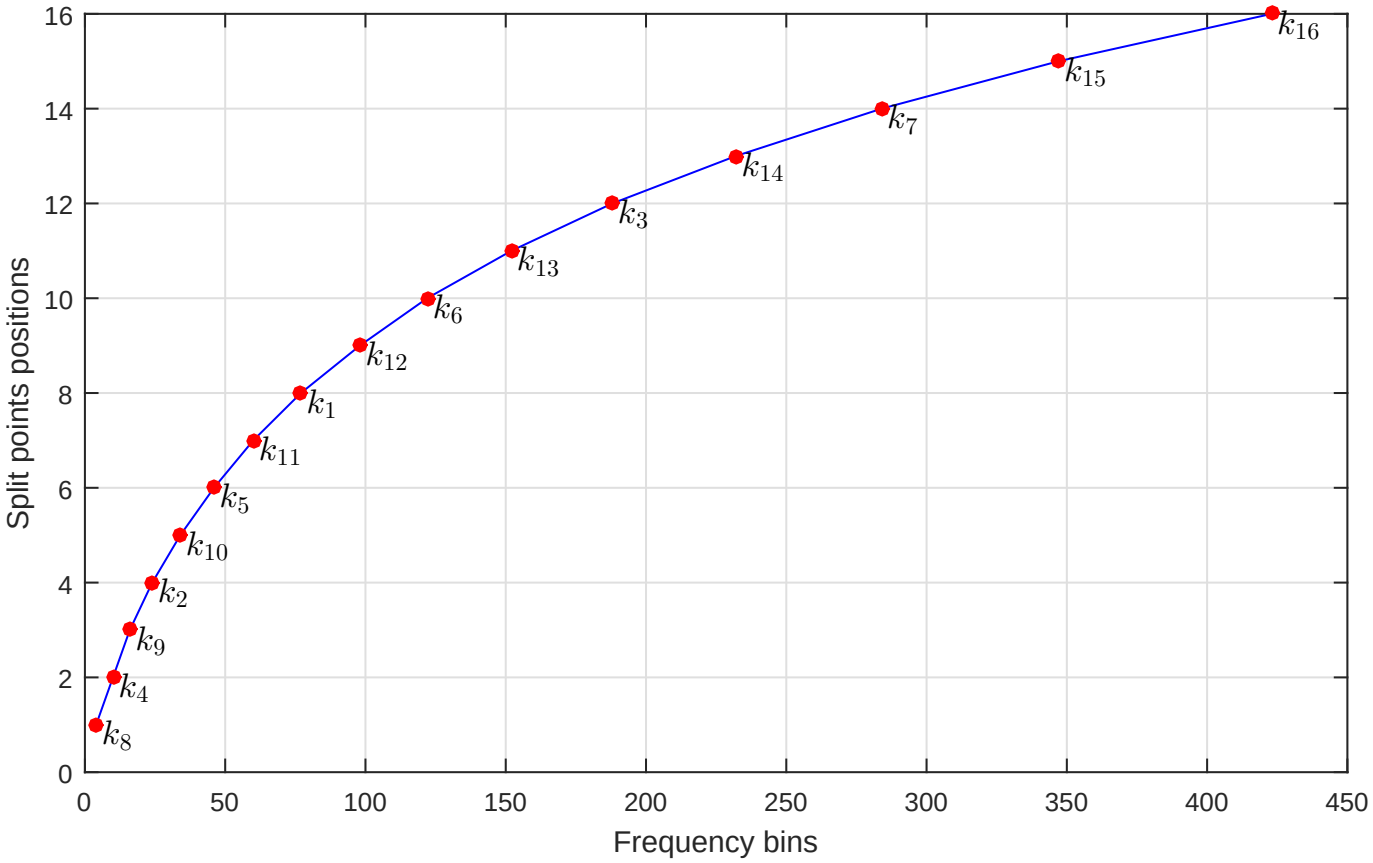


Figure 4.9: Position of the split points on mel weighted frequency bins with sample rate 32 kHz and windows length 32ms.

points, which will be our split points, are mel weighted. Figure 4.9) is an example mel weighted frequency bins used for split points and the position of the split points, for number of split points $M = 16$.

As it was mentioned before our implementation is similar to 4.1.2 in which a recursive level wise structure is followed. In 4.1.2, the first split point k_1 position is determined based on maximum variance of the signal. For other split points same process is done recursively in their own bandwidth which ,in turn, depends on their parent split point position. In the course of our thesis, the first split point k_1 is indeed the middle point of mel scaled frequency points derived from Equation (4.6). For the next level the same recursive process happening in 4.1.2, is done, but split points position are based on being the middle point in their sub-bands (see Figure 4.5).

For our thesis work, we chose number of split points 16 and 24 for further investigation. This helps to have same environment to compare with LPC. From now we will refer to DQ envelope modeling with 24 split points with "DQ 24" and DQ envelope modeling with 16 split points with "DQ 16".

Quantization accuracy for Uniform Quantization and Weighting for Vector Quantization

In Uniform Quantization we need to define quantization level on which quantization accuracy is based. In [24] each spilt point's quantization level depends on its bandwidth. The equation used for quantization accuracy is defined as [24]:

$$ql_k = \frac{BW_k}{BW_{frame}}, \tag{4.7}$$

where ql_k, is the quantization level of split point k, BW_k is the bandwidth of the split point k and BW_{frame} is the bandwidth of the frame. Obviously, for the first split point we ql would be equal to 1 and for the other split points it would $0 \leqslant ql_k \leqslant 1$. Multiplying an arbitrary constant to ql_k the quantization accuracy will be derived. Our implementation is similar to [24]. However, since the whole process of DQ is a level wise process and sibling split points (split points in the same level) have same impact on their children (split points in one level lower), we thought to have same quantization accuracy for split points located in the same level. To do so, first ql_k, of each split point is computed using Equation (4.7). Afterwards, for each level the mean of ql_ks of split points located in that level, will be computed. The result would be the quantizaion level of each split point located in that level. To make the process more clear,in Figure 4.10 we compute quantization level of each split point located in the last level (circled with a blue line). First using Equation (4.7) we compute each split points quantization level. We then take the mean of the resulted quantization level:

$$qll = \frac{\sum_{i=1}^{m} ql_i}{m}, \tag{4.8}$$

where qll is the quantization level of that level, m is the number of split points in that level. qll will be the quantization level of each split point located in that level.

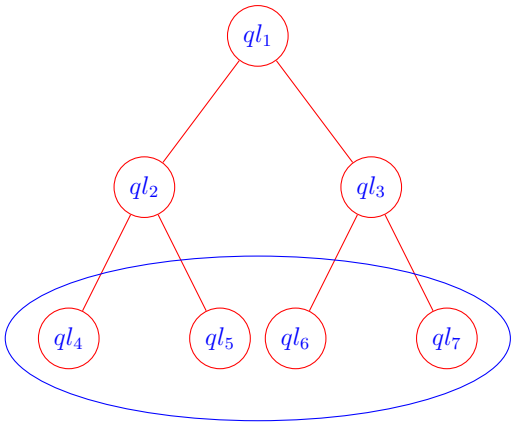


Figure 4.10: Example of quantization level structure with number of split points 7.

Quantization accuracy will be $q_{accuracy} = S * qll$, where S is an arbitrary constant whose value in our case is chosen in a way that bit consumption would not exceed 39 bits.

Weighting in Vector Quantization is also derived from Equation (4.9). However, qll which are between 0 and 1, is not approporiate to use as a weighting coefficients in Vector Quantization. So, we normalize it in a way that the minimum value for weighting would be one, that is:

$$w_i = qll_i / qll_{lastlavel}. \tag{4.9}$$

4.3 Performance Analysis

Before starting the objective and subjective measurements, we visually inspected the spectral envelope derived from DQ to confirm positioning of the split points have been done appropriately. To do so, we checked some random frames with their envelopes. One random frame with the envelope is depicted in Figure 4.11. From the figure we can see that in lower frequencies due to congestion of split points, the envelope derived from DQ, models harmonics too. Consequently, we do not get as smooth envelope as we would like. This analysis shows that the initial assumption claiming: the more number of split points, the more accurate spectral envelope is obtained does not apply when the number of split points are high.

By looking at the positions of the split points in DQ 24 and DQ 16, we realized that the distance between between split points is less than 300 Hz (9 frequency bins with bin resolution 31.25 Hz) in the first 10 and 6 split points respectively. Therefore, we decided to have a minimum distance threshold between the split points. By trying different values for mentioned threshold the best result in terms of the envelope and LSD was obtained with minimum distance around 430 Hz (14 bin with bin resolution 31.25 Hz). In Figure 4.12 which is the same frame in Figure 4.11, it can be seen that the envelope is smoother than Figure 4.11.

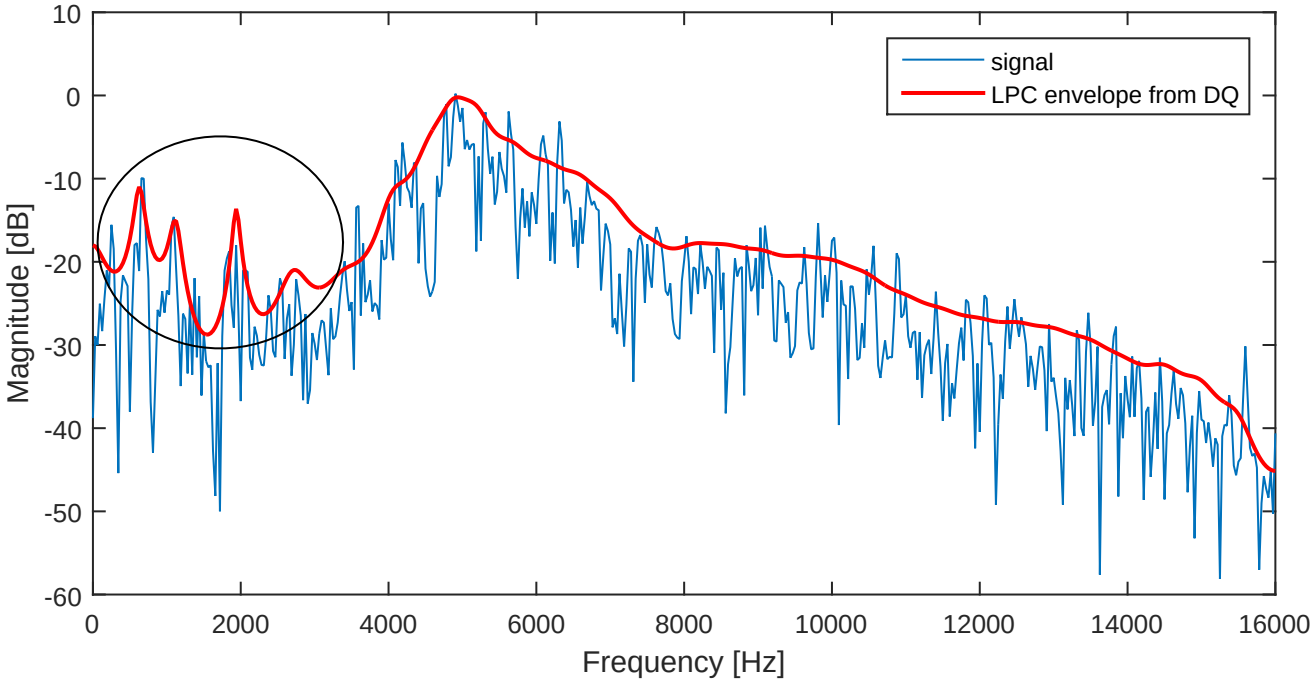


Figure 4.11: A frame with DQ envelope modeling following mel scale without minimum distance threshold.

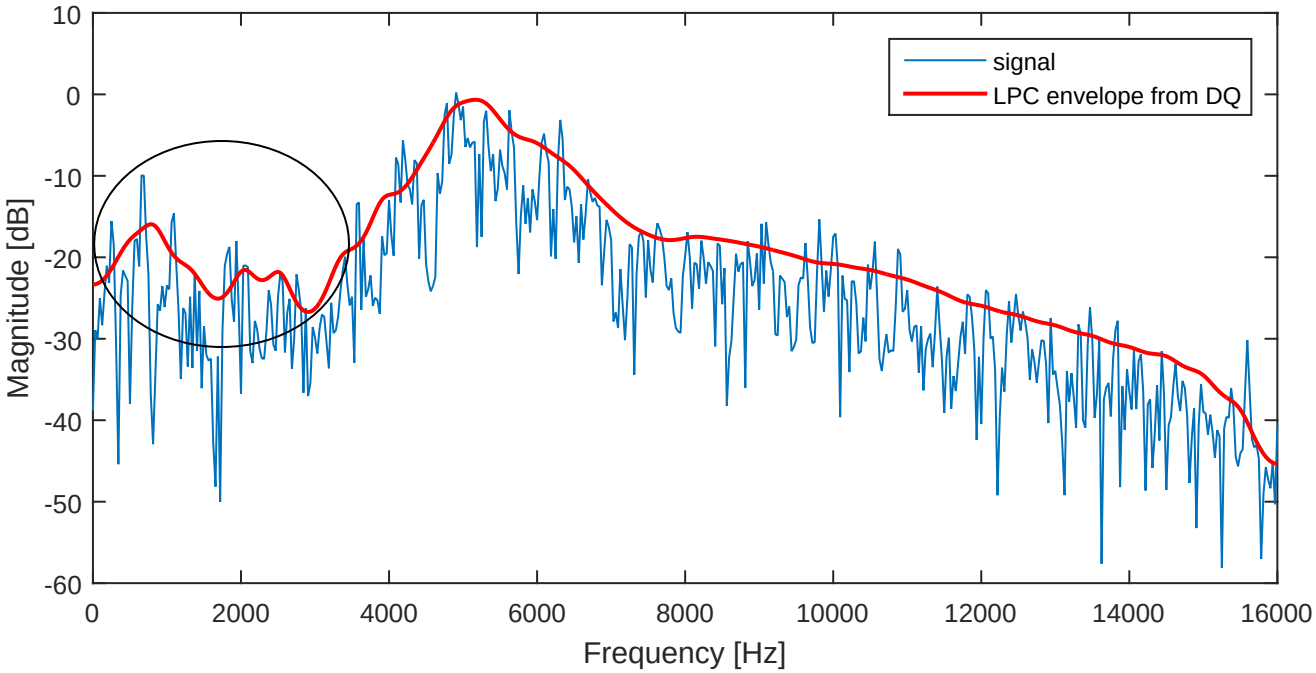


Figure 4.12: A frame with DQ envelope modeling following mel scale with minimum distance threshold 430 Hz.

4.4 Experiments and Results

To evaluate the performance of DQ 24 and DQ 16 we use same measuring tools: SD, POLQA and MUSHRA test which we used for LPC. However, in DQ technique, split points are positioned based on mel scale. Hence, the weighting SD is based on mel scale. We also evaluate uniform quantizer by SD measurement.

4.4.1 SD

Results of different SD measurements are presented in table 4.1. From the table 4.1a and 4.1b we can see that , unquantized DQ 24 has significantly better performance than unquantized DQ 16 (about 0.4 dB). But, this difference becomes less than 0.1 dB when DQ 24 and DQ 16 are quantized by vector quantizer (VQ). This degradation in the performance of DQ 24 could result from the shortage of bits in VQ to quantize 24 DQ parameters(γ_k).

In the case of uniform quantization, table shows that DQ 16 results in better SD compared to DQ 24. In uniform quantization, we set the quantizer accuracy such that the bit consumption be around 39 so that we have same experiment environment as vector quantization case. Consequently, in DQ 24 the quantization accuracy must be set to lower value compared to DQ 16. That is why, DQ 24 with uniform quantization has worse performance than DQ 16 with uniform quantization.

Last but not least, in all measurements of DQ 16, uniform quantization results in lower SD and weighted SD, compared to vector quantization. The difference is more significant when the reference is unquantized DQ 16. This confirms the fact that in DQ 16, DQ parameters (γ_k) are uncorrelated with each other and can be quantized independently.

4.4.2 POLQA and MUSHRA

The first POLQA test compares: FB-CELP DQ 24, which is quantized by VQ and integrated into full-band extended version of CELP speech coding algorithm, FB-CELP LPC without weighting obtained in section 3.7, ACELP+SBR and TCX+SBR speech codecs. Figure 4.13 shows that for both clean and noisy speech items FB-CELP DQ 24 has worse performance compared to FB-CELP LPC. This difference will be more significant in case of FB-CELP LPC with weighting and new perceptual model.

We set up an informal MUSHRA listening test with 12 speech items and two listeners. Figure 4.14 shows the difference MUSHRA scores. The difference is calculated with respect to the condition FB-CELP DQ 24 (FB_CELP_DQ). We can see that our observation from the POLQA test is confirmed by MUSHRA listening test. The reason why DQ 24 has poor performance could be that in vector quantizer 39 bits is not enough to quantize 24 parameters (γ_k) of DQ.

Table 4.1: SD results DQ 24 versus DQ 16.

	SD	Outlier 2−4 dB[%]	Outlier > 4 dB[%]	Bit Rate
DQ 24 no Q	2.5956	73.98	3.74	
DQ 24 uniform Q	3.5157	77.59	22.06	39.13
DQ 24 VQ	3.6223	56.31	34.22	39
DQ 16 no Q	3.0877	71.51	16.58	
DQ 16 uniform Q	3.6086	68.42	30.78	39.34
DQ 16 VQ	3.6566	58.38	36.53	39

(a) SD of DQ with reference true envelope.

	SD	Outlier 2−4 dB[%]	Outlier > 4 dB[%]
DQ 24 uniform Q	2.3713	52.87	0
DQ 24 VQ	2.1721	44.15	3.25
DQ 16 uniform Q	1.7873	13.17	0.07
DQ 16 VQ	2.2558	47.63	6.30

(b) SD of DQ with reference unquantized DQ.

	weighted SD	Outlier 2−4 dB[%]	Outlier > 4 dB[%]	Bit Rate
DQ 24 no Q	2.6911	78.40	4.70	
DQ 24 uniform Q	3.5329	77.97	21.65	39.13
DQ 24 VQ	3.6148	52.85	28.17	39
DQ 16 no Q	3.0891	72.47	15.53	
DQ 16 uniform Q	3.3965	74.08	22.96	39.49
DQ 16 VQ	3.6559	56.18	36.53	39

(c) Mel weighted SD with Reference True Envelope.

	weighted SD	Outlier 2−4 dB[%]	Outlier > 4 dB[%]
DQ 24 uni Q	2.3035	86.29	0
DQ 24 VQ	1.9022	32.65	2.05
DQ 16 uni Q	1.4029	0.48	0
DQ 16 VQ	2.1871	44.16	6.08

(d) Mel weighted SD of DQ with reference unquantized DQ.

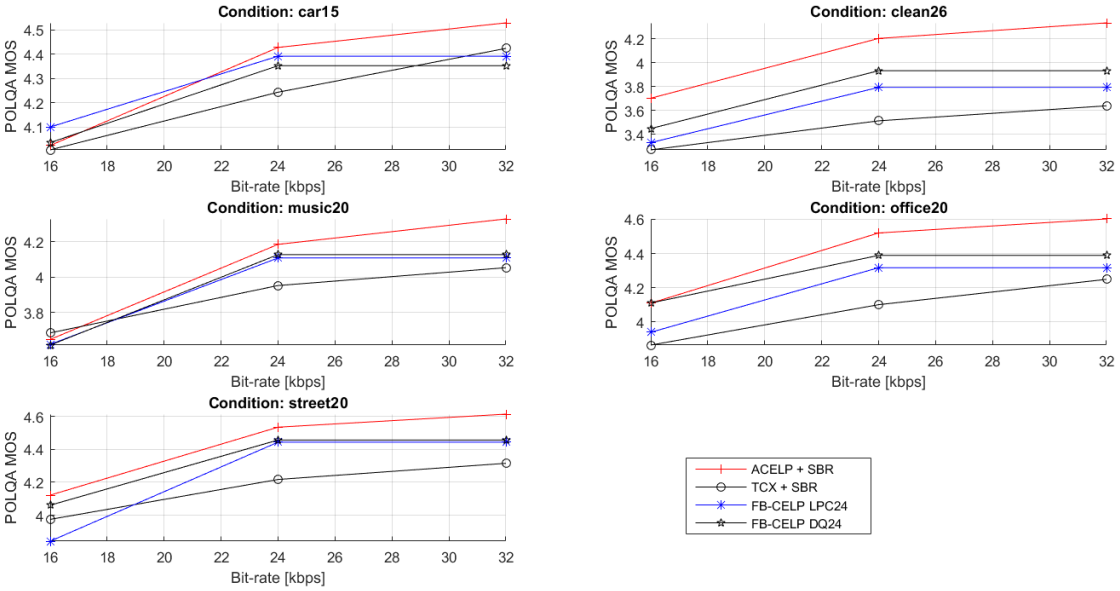


Figure 4.13: Results of POLQA test. FB-CELP DQ 24 vs FB-CELP LPC obtained in section 3.7.

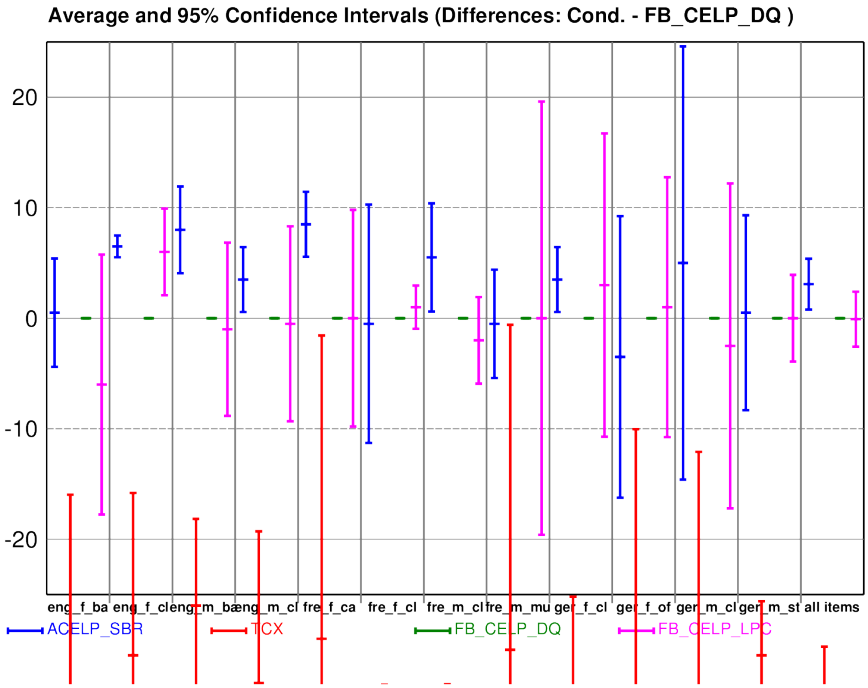


Figure 4.14: Results of informal MUSHRA listening test. FB-CELP DQ 24 vs FB-CELP LPC obtained in section 3.7.

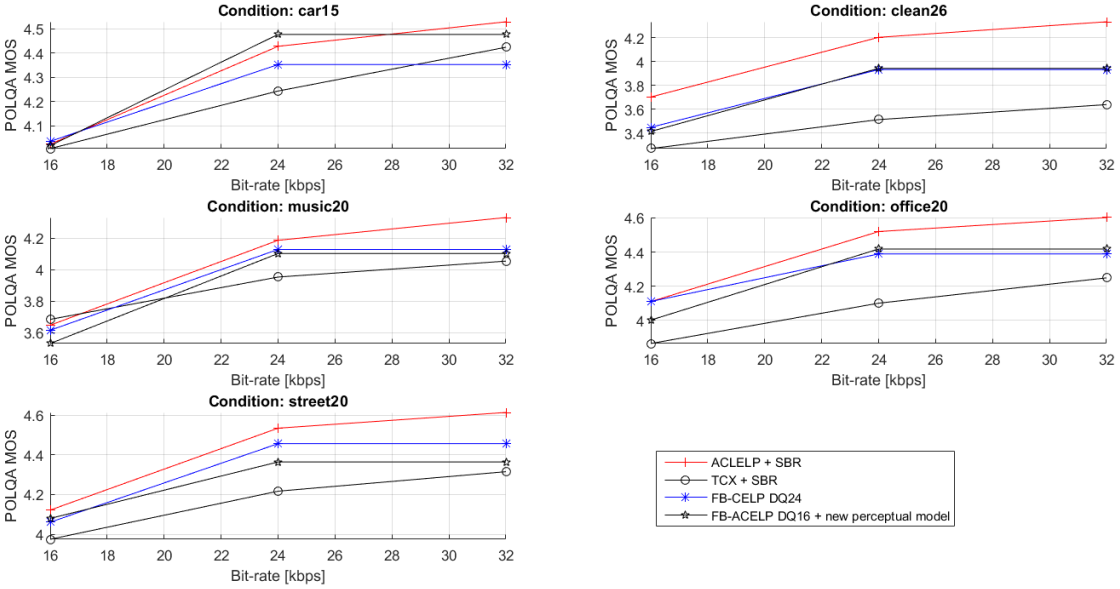


Figure 4.15: Results of POLQA test. FB-CELP DQ 16 vs FB-CELP DQ 24 obtained in section 3.7.

Another investigation is comparison between DQ 16 (with new perceptual model obtained in section 3.10) and DQ 24 using POLQA test. From the Figure 4.15 we can see in most cases FB-CELP DQ 16 results in either better or equal quality compared to FB-CELP DQ 24. Since we have seen the performance of FB-CELP DQ 24 in MUSHRA test, we chose FB-CELP DQ 16 with new perceptual model for the final assessment.

5 Final Evaluation

In this chapter we present the final assessment of the envelope models derived and extended from LPC and DQ techniques. Chapter starts with the objective evaluation of all possible envelope models obtained in chapters 3 and 4. Finally, a subjective assessment is performed by following the subjective evaluation methodology MUSHRA. The two full-band speech coding schemes using DQ 16 and the weighted LPC are compared to each other and against the speech coding mode and one of the music coding mode of the state-of-the-art coding standard ISO/MPEG USAC [27].

5.1 Overall Objective Assessment

To have an overall assessment of obtained spectral envelopes, we set a POLQA test where we compare: ACELP+SBR, TCX+SBR, FB-CELP DQ 24, FB-CELP DQ 16, FB-CELP DQ 16 with new perceptual model, FB-CELP LPC, FB-CELP LPC with new perceptual model and weighted LPC with new perceptual model. For this test we have several clean speech items. Figure 5.1 shows that weighted LPC with new perceptual model has the best performance among the obtained envelope models. Hence, we choose weighted LPC with new perceptual model for the final MUSHRA test. The second best result corresponds to DQ 16. Nevertheless, we choose DQ 16 along with new perceptual model for the final MUSHRA test, as new perceptual model has been found through informal listening tests to result in better perceived speech signal.

5.2 Pre-final Objective Assessment

Before doing the final listening test, we compare weighted LPC and DQ 16 using SD and POLQA measurements.

Table 5.1 shows that DQ 16 with uniform quantization is quite close to LPC. Integration of DQ with uniform quantization into ACELP is out of the scope of this thesis. Otherwise it would be interesting to evaluate the performance of uniform quantization in the speech codec in full-band case. Generally, we can see the superiority of LPC over DQ which can be consequence of optimal weighting in LPC.

POLQA score are depicted in Figure 5.2. In most cases, FB-CELP DQ 16 leads to worse results than FB-CELP weighted LPC. Similar to SD result, this difference could be because of the weighting which in DQ, is not as optimal as weighting in LPC.

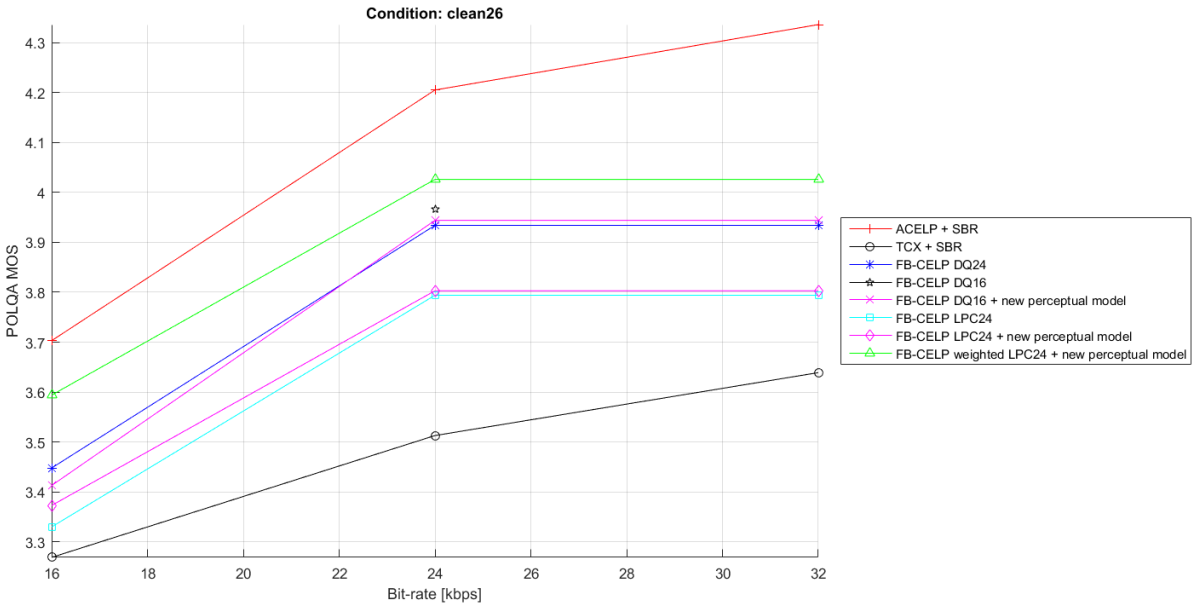


Figure 5.1: POLQA test for all obtained envelope models using clean speech items.

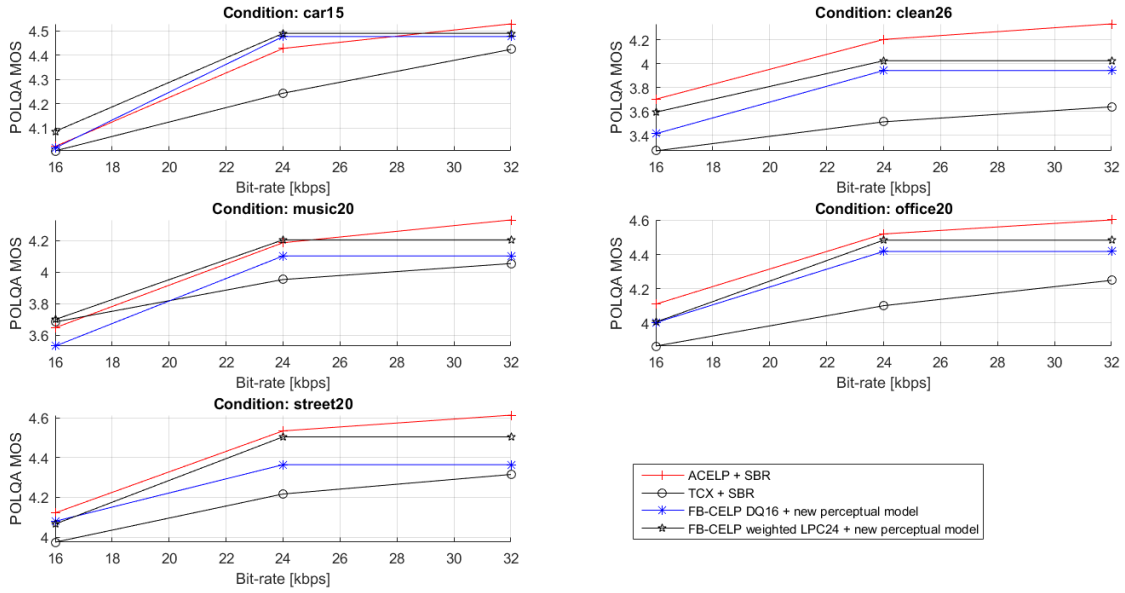


Figure 5.2: POLQA for DQ 16 and weighted LPC.

Table 5.1: SD results weighted LPC versus DQ 16.

	SD	Outlier 2−4 dB[%]	Outlier > 4 dB[%]	Bit Rate
DQ 16 uniform Q	3.6086	68.42	30.78	39.34
DQ 16 VQ	3.6566	58.38	36.53	39
weighted LPC VQ	3.5930	72.87	26.26	39

(a) SD of DQ and LPC with reference true envelope

	SD	Outlier 2−4 dB[%]	Outlier > 4 dB[%]
DQ 16 uniform Q	1.7873	13.17	0.07
DQ 16 VQ	2.2558	47.63	6.30
weighted LPC VQ	1.9805	43.25	0.17

(b) SD of DQ and LPC with reference unquantized DQ or LPC

	weighted SD	Outlier 2−4 dB[%]	Outlier > 4 dB[%]	Bit Rate
DQ 16 uniform Q	3.3965	74.08	22.96	39.49
DQ 16 VQ	3.6559	56.18	36.53	39

(c) Mel weighted SD with Reference true envelope

	weighted SD	Outlier 2−4 dB[%]	Outlier > 4 dB[%]	Bit Rate
weighted LPC VQ	4.1367	53.71	41.21	39

(d) Bark weighted SD with reference true envelope

	weighted SD	Outlier 2−4 dB[%]	Outlier > 4 dB[%]
DQ 16 uni Q	1.4029	0.48	0
DQ 16 VQ	2.1871	44.16	6.08

(e) Mel weighted SD of DQ with reference unquantized DQ

	weighted SD	Outlier 2−4 dB[%]	Outlier > 4 dB[%]
weighted LPC VQ	1.1802	8.04	0.28

(f) Bark weighted SD of LPC with reference unquantized LPC

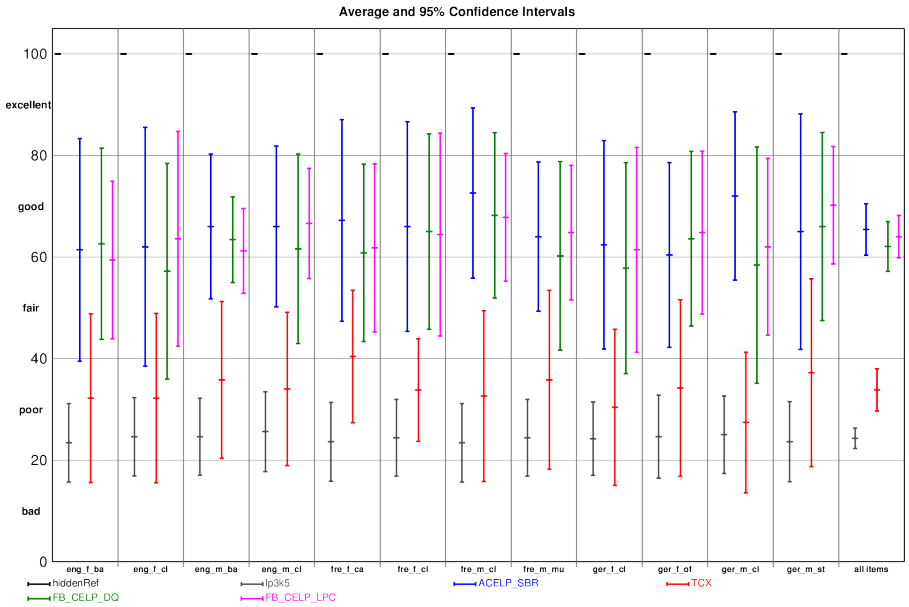


Figure 5.3: Average absolute MUSHRA scores for 12 speech items with 8 listeners using 95% confidence intervals of Student's t-distribution.

5.3 Subjective Assessment

In the final MUSHRA listening test, we have same setup as section 3.8.2: 12 speech items consisting of 6 noisy and 6 clean speech in German, English and French languages, and 8 listeners including 6 expert listeners. We analyse the results of MUSHRA test by using student's t-test with 95% confidence interval.

In addition to weighted FB-CELP LPC (FB_CELP_LPC) and FB-CELP DQ 16 (FB_CELP_DQ), listeners were asked to evaluate the hidden reference, the 3.5 kHz filtered reference used as anchor, the standardized ACELP+SBR from ISO/MPEG USAC (ACELP_SBR) and the standardized TCX+SBR from ISO/MPEG USAC (TCX).

The average absolute MUSHRA scores are depicted in Figure 5.3. We can observe that weighting and new perceptual model have outstandingly improved the performance of FB-CELP LPC. In most items, LPC performs as good as ACELP+SBR. Figure also shows that FB-CELP DQ 16 is slightly worse than weighted LPC but the results of FB-CELP DQ 16 lie in the "good" condition.

The difference MUSHRA scores are plotted in Figure 5.4, Figure 5.5 and Figure 5.6 that respectively correspond to clean speech items, noisy speech items and all speech items. The reason why we have separated analysis of clean and noisy is that clean items and noisy items have different characteristics. In Figure 5.4 FB-CELP LPC is significantly worse than ACELP+SBR in item $german \quad male \quad clean$ and FB-CELP DQ 16 is significantly worse than ACELP+SBR in 3 items out of 6 items. Figure 5.5 shows that in one item ($german \quad female \quad office$) FB-CELP LPC is significantly

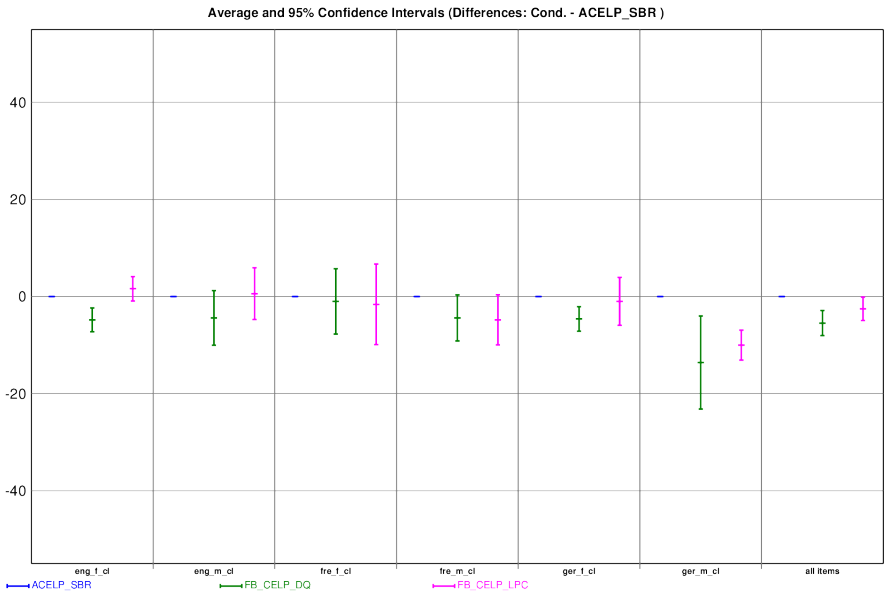


Figure 5.4: Difference MUSHRA scores for 6 clean speech items with 8 listeners using 95% confidence intervals of Student's t-distribution. The difference is computed with SBR.

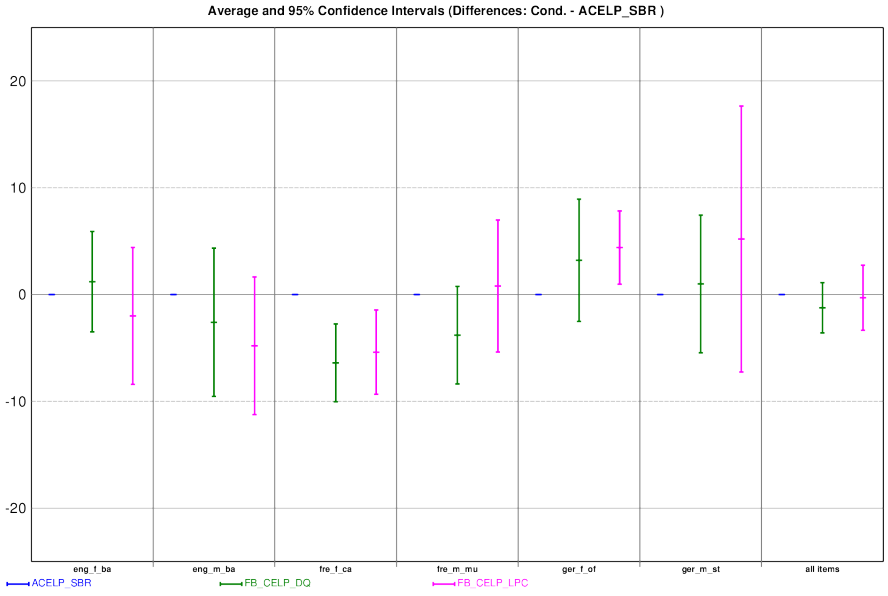


Figure 5.5: Difference MUSHRA scores for 6 noisy speech items with 8 listeners using 95% confidence intervals of Student's t-distribution. The difference is computed with SBR.

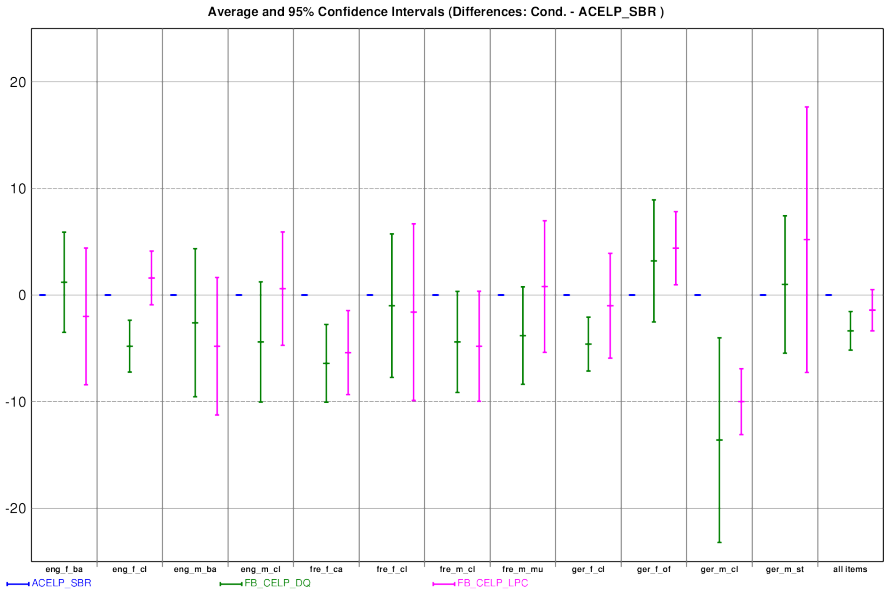


Figure 5.6: Difference MUSHRA scores for 12 speech items with 8 listeners using 95% confidence intervals of Student's t-distribution. The difference is computed with SBR.

better than ACELP+SBR ,while in item $french \quad female \quad car$ ACELP+SBR is significantly better than FB-CELP LPC and FB-CELP DQ16. For most of noisy items FB-CELP DQ 16 performs slightly worse than LPC and SBR. Over all items, weighted LPC has statistically better performance than FB-CELP DQ 16 and there is no statistical difference between FB-CELP LPC and ACELP+SBR.

6 Conclusion

6.1 Summary of Our Work

In this thesis, we studied the spectral envelope modelling for full-band speech coding. Two spectral envelope modelling techniques LPC and DQ were investigated. To compare LPC and DQ, same framework based on CELP coding scheme was used. In addition, same vector quantization technique along with same objective and subjective measurements were applied on both techniques to assess their performance.

For LPC, the different parameters found in conventional narrow-band and wide-band spectral modelling, like the LPC order, the pre-emphasis factor, and the perceptual weighting factor were optimized through different optimization processes. MA-MSVQ parameters were then determined using SD and bark weighted SD measurements. The estimated LPC along with the quantizer was integrated into a full-band extended version of CELP speech coding algorithm for evaluation of the overall quality of the speech codec. Results of objective and subjective measurements showed that proposed LPC was significantly inferior to the state-of-the-art wideband speech coder ACELP extended with the bandwidth extension technique SBR. Consequently, we improved the estimated LPC by applying bark-based weighting coefficients along with estimation of perceptual model parameters. Objective measurements SD and POLQA showed that weighting and new perceptual model improved performance of the estimated envelope model and the overall quality of the speech codec. Hence, we chose LPC with weighting and new perceptual model for the final assessment.

For derivation of spectral envelope model using DQ, different metrics had to be addressed. First challenge was positioning of DQ parameters throughout the bandwidth. We employed fixed positioning scheme based on mel scaling to locate 24 DQ parameters. By performance analysis, we observed the congestion of DQ parameters in low frequencies such that the spectral envelope models harmonics as well. Therefore, we forced minimum distance threshold between DQ parameters. SD measurement confirmed necessity of threshold imposing. An informal listening test showed DQ 24 performs worse than LPC without weighting and new perceptual model. It was assumed that bit allocation in vector quantization is not enough for quantizing 24 DQ parameters. Hence, we changed the number of parameters from 24 to 16. Comparison between DQ 24 and DQ 16 using SD and POLQA showed that DQ 16 is almost as good as DQ 24. The results confirmed our initial assumption about lack of bits in VQ to quantize 24 parameters of DQ. Since we had already had listening test on DQ 24 we chose DQ 16 for the final assessment. We also evaluated DQ 24 and DQ 16 along with uniform quantization. According to the results, uniform quantization performs somewhat better than VQ.

Finally, both techniques were compared within a complete full-band speech coding scheme. The comparison was based on objective and subjective evaluation. The final listening test showed that LPC and DQ deliver the same range of quality overall although, LPC slightly outperform DQ. It can be explained by a sub-optimal design

of the quantization of DQ parameters. Indeed, with the current design an order of 16 was found optimal, which seems too low compared to the LPC order and may indicate that the bit allocation for the parameters is insufficient for reaching higher orders. Another observation was that the proposed full-band spectral modelling corresponded to LPC performs only slightly worse for clean speech while being on par for noisy speech in comparison with the state-of-the-art codec ISO/MPEG Unified Speech and Audio Coder (USAC) using a wide-band speech coder based on ACELP and Bandwidth extension based on SBR.

6.2 Future Work

This work can be taken forward in different directions such as:

- In DQ technique, weighting algorithm can be optimized using psychoacoustical scaling such as mel and bark scaling techniques. However, since DQ parameters are not quantized from low frequency components to high frequency components, finding an appropriate weighting scheme for DQ is more complicated than weighting in LPC.
- Investigation on entropy coding using DQ was out of the scope of this work. As part of the future work, DQ along with entropy coding can be assessed within a full-band speech coding scheme.
- In this thesis we wanted to have fair comparison between LPC and DQ. Consequently, we used same VQ for both techniques. However, measurements showed that the performance of DQ 24 can be improved by more bit allocation in VQ. Derivation of new VQ with higher number of bits and different constellation can be surveyed in the future.
- LPC showed a great performance but it is at the first stage of development. Further engineering tuning and optimization can be applied on LPC.

References

[1] 3GPP. TS 26.190; Adaptive Multi-Rate (AMR-WB) speech codec. 3GPP, 2007.

[2] 3GPP. TS 26.445; EVS Codec Detailed Algorithmic Description. 3GPP, 2014.

[3] J. P. Adoul, P. Mabilleau, M. Delprat, and S. Morissette. Fast celp coding based on algebraic codes. In *Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP '87.*, volume 12, pages 1957–1960, 1987.

[4] Marwan Al-Akaidi. *Fractal Speech Processing.* Cambridge University Press, 2010.

[5] B. Atal. Predictive coding of speech at low bit rates. *IEEE Transactions on Communications*, 30:600–614, 1982.

[6] V. Atti and A. Spanias. A simulation tool for introducing algebraic celp (acelp) coding concepts in a dsp course. In *Digital Signal Processing Workshop, 2002 and the 2nd Signal Processing Education Workshop. Proceedings of 2002 IEEE 10th*, pages 306–311, 2002.

[7] T. Bäckström. *Speech Coding with Code Excited Linear Prediction.* In press, Springer, 2017.

[8] B. Bessette, R. Lefebvre, and R. Salami. Universal speech/audio coding using hybrid acelp/tcx techniques. In *Proceedings. (ICASSP '05). IEEE International Conference on Acoustics, Speech, and Signal Processing, 2005.*, 2005.

[9] M. Caetano and X. Rodet. Improved estimation of the amplitude envelope of time-domain signals using true envelope cepstral smoothing. In *2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4244–4247, 2011.

[10] R. V. Cox, S. F. De Campos Neto, C. Lamblin, and M. H. Sherif. Itu-t coders for wideband, superwideband, and fullband speech communication [series editorial]. *IEEE Communications Magazine*, 47(10):106–109, 2009.

[11] Sneha Das. Source modelling based on higher-order statistics for speech enhancement applications. Master's thesis, Friedrich-Alexander-Universit, 2016.

[12] Fraunhofer Institute for Integrated Circuits IIS. Extended HE-AAC – Bridging the gap between speech and audio coding. Fraunhofer IIS, 2013.

[13] Thierry Galas and Xavier Rodet. An improved cepstral method for deconvolution of source filter systems with discrete spectra: Application to musical sound signals. In *Proceedings of the International Computer Music Conference (ICMC)*, pages 82–84, 1990.

[14] W. R. Gardner and B. D. Rao. Theoretical analysis of the high-rate vector quantization of lpc parameters. *IEEE Transactions on Speech and Audio Processing*, 1995.

[15] Allen Gersho and Robert M. Gray. *Vector Quantization and Signal Compression (The Springer International Series in Engineering and Computer Science)*. Springer, 1991.

[16] ITU. P.862 : Perceptual evaluation of speech quality (PESQ). ITU, 2001.

[17] ITU. G.719 : Low-complexity, full-band audio coding for high-quality, conversational applications. ITU, 2008.

[18] ITU. BS.1534-2 : Method for the subjective assessment of intermediate quality level of audio systems. ITU, 2014.

[19] ITU. P.863 : Perceptual Objective Listening Quality Assessment. ITU, 2014.

[20] T. Jähnel, T. Bäckström, and B. Schubert. Envelope modeling for speech and audio processing using distribution quantization. In *Signal Processing Conference (EUSIPCO), 2015 23rd European*, pages 584–588, 2015.

[21] Peter Kabal, Ravi, and Prakash Ramachandran. The computation of line spectral frequencies using chebyshev polynomials. In *Proc. IEEE ht. Conf. on Acoustics, Speech, Signal Pre cessing*, pages 1419–1426, 1986.

[22] W.B. Kleijn and K.K. Paliwal, editors. *Speech Coding and Synthesis*. Elsevier Science, 12 1995.

[23] A. M. Kondoz. *Digital Speech Coding for Low Bit Rate Communication Systems*. John Wiley & Sons Ltd, 2004.

[24] Srikanth Korse, Tobias Jähnel, and Tom Bäckström. Entropy coding of spectral envelopes for speech and audio coding using distribution quantization. *Interspeech 2016*, pages 2543–2547, 2016.

[25] J. Makhoul. Correction to "linear prediction: A tutorial review". *Proceedings of the IEEE*, 64(2):285–285, 1976.

[26] A.V. McCree. Quantization of linear prediction coefficients using perceptual weighting, 2005.

[27] Max Neuendorf, Markus Multrus, Nikolaus Rettelbach, Guillaume Fuchs, Julien Robilliard, Jérémie Lecomte, Wilde Stephan, Stefan Bayer, Sascha Disch, Christian Helmrich, et al. The iso/mpeg unified speech and audio coding standard—consistent high quality for all content types and at all bit rates. *Journal of the Audio Engineering Society*, 61(12):956–977, 2013.

[28] K. K. Paliwal and B. S. Atal. Efficient vector quantization of lpc parameters at 24 bits/frame. *IEEE Transactions on Speech and Audio Processing*, 1(1):3–14, 1993.

[29] John G. Proakis and Dimitris K Manolakis. *Digital Signal Processing (4th Edition)*. Pearson, 2006.

[30] Saito. *Fundamentals of Speech Signal Processing*. Academic Press, 1986.

[31] M. Schroeder and B. Atal. Code-excited linear prediction(celp): High-quality speech at very low bit rates. In *Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP '85.*, volume 10, pages 937–940, 1985.

[32] M. Schroeder and B. Atal. Code-excited linear prediction(celp): High-quality speech at very low bit rates. In *Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP '85.*, volume 10, pages 937–940, 1985.

[33] Diemo Schwarz and Xavier Rodet. Spectral envelope estimation and representation for sound analysis–synthesis. 2009.

[34] F. Soong and B. Juang. Line spectrum pair (lsp) and speech data compression. In *Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP '84.*, volume 9, pages 37–40, 1984.

[35] F. K. Soong and B. H. Juang. Optimal quantization of lsp parameters [speech coding]. In *Acoustics, Speech, and Signal Processing, 1988. ICASSP-88., 1988 International Conference on*, pages 394–397 vol.1, 1988.

[36] S. Umesh, L. Cohen, and D. Nelson. Fitting the mel scale. In *Acoustics, Speech, and Signal Processing, 1999. Proceedings., 1999 IEEE International Conference on*, volume 1, pages 217–220 vol.1, 1999.

[37] F. Villavicencio, A. Robel, and X. Rodet. Improving lpc spectral envelope extraction of voiced speech by true-envelope estimation. In *2006 IEEE International Conference on Acoustics Speech and Signal Processing Proceedings*, volume 1, 2006.

[38] F. Villavicencio, A. Robel, and X. Rodet. All-pole spectral envelope modelling with order selection for harmonic signals. In *2007 IEEE International Conference on Acoustics, Speech and Signal Processing - ICASSP '07*, volume 1, pages I–49–I–52, 2007.